ESCUELA SUPERIOR DE INGENIERÍA

MÁSTER EN SEGURIDAD INFORMÁTICA
(CIBERSEGURIDAD)

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

AUTOR: CARLOS CARRETERO AGUILAR

Cádiz, julio 2018

Escuela Superior de Ingeniería

ESCUELA SUPERIOR DE INGENIERÍA

MÁSTER EN SEGURIDAD INFORMÁTICA
(CIBERSEGURIDAD)

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

DIRECTOR: CARLOS RODRÍGUEZ CORDÓN
AUTOR: CARLOS CARRETERO AGUILAR

Cádiz, julio 2018

Agradecimientos

A Carlos, por su inestimable predisposición a ayudar en todo lo posible.

A Manolo, por sus inagotables ganas de ayudarme en mi desarrollo profesional.

A mis padres, por darme la oportunidad de crecer como persona y convertirme en quién soy.

A las tres mujeres más bonitas del mundo, por darme todo su amor.

A todos mis compañeros del Máster, por hacerme pasar un año inolvidable.

A todos los que han participado de alguna manera u otra, gracias

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

REF: 0000002

ÍNDICE

CLIENTE: EMPRESA PROVINCIAL DE INFORMACIÓN DE CÁDIZ S.A. (EPICSA)
PLAZA MADRID S/N, EDIFICIO CARRANZA,
FONDO SUR, LOCAL 10, 11010 CÁDIZ
956261500

AUTOR: CARLOS CARRETERO AGUILAR
INGENIERO INFORMÁTICO
25603515-F
CARLOS.CARRETEROAGUILAR@ALUM.UCA.ES

FIRMADO:
SOLICITANTE AUTOR

CÁDIZ, A 1 DE JULIO DE 2018

Índice de contenido

Índice de figuras

Índice de tablas

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

REF: 0000002

MEMORIA

CLIENTE: EMPRESA PROVINCIAL DE INFORMACIÓN DE CÁDIZ S.A. (EPICSA)
PLAZA MADRID S/N, EDIFICIO CARRANZA,
FONDO SUR, LOCAL 10, 11010 CÁDIZ
956261500

AUTOR: CARLOS CARRETERO AGUILAR
INGENIERO INFORMÁTICO
25603515-F
CARLOS.CARRETEROAGUILAR@ALUM.UCA.ES

FIRMADO:
SOLICITANTE AUTOR

CÁDIZ, A 1 DE JULIO DE 2018

1 Objeto

El objetivo de este proyecto es diseñar y desplegar una red honeynet virtual que sirva como punto de detección y análisis de ataques informáticos en la frontera de la red de la Diputación de Cádiz. Dicha honeynet debe constituirse de un conjunto de equipos intencionadamente vulnerables interconectados entre sí. La honeynet a diseñar y desplegar debe proveer de mecanismos de detección de ataques, de las herramientas necesarias para el control de los equipos vulnerables, así como de un sistema de visualización de todos los datos recogidos por la honeynet.

La red honeynet debe proveer de la posibilidad de desplegar todos los sistemas y procesos vulnerables necesarios dentro de una misma máquina física, de tal manera que se optimice al máximo la facilidad de despliegue, de mantenimiento y de portabilidad.

2 Antecedentes

La importancia de la detección y análisis de ataques informáticos en el mundo moderno está fuera de toda duda. Hoy en día, la gran interconexión de las grandes organizaciones y el acceso de la población a Internet provocan un gran tráfico de información cada segundo. Según el Centro Criptográfico Nacional [1], en el año 2016, se registraron 3.675.824.813 usuarios activos en Internet, lo que supone la mitad de la población mundial actual.

Toda esta información no está libre de ser atacada, de hecho, hoy en día, el número de ataques informáticos que se producen aumentan a un ritmo vertiginoso., los ataques informáticos suponen actualmente una amenaza mundial del mismo nivel que el desempleo o las crisis económicas.

En el año 2016, los elementos más significativos en la ciberseguridad fueron:

- Ciberespionaje de naturaleza económica y/o política.
- Ciberdelincuencia.
- Cifrado de la información.
- Ciberactivismo.
- Script Kiddies.
- Ataques DDoS.

Según un estudio realizado por la empresa Accenture [2], cada año se producen, de media, 130 brechas de seguridad en grandes empresas, sin contar todas aquellas de menor calado, produciéndose un incremento en un 27.4% del número de brechas de seguridad que se producen anualmente. Se calcula un valor medio del coste de sufrir una brecha de seguridad en una gran empresa de unos 2,4 millones de dólares, con un tiempo medio de 50 días para solucionar y recuperarse del ataque informático sufrido. Además, mediante una encuesta realizada por la misma empresa en colaboración con Ponemon Institute [3], el coste, en millones de dólares, de los ataques informáticos, en 2017 por país fue:

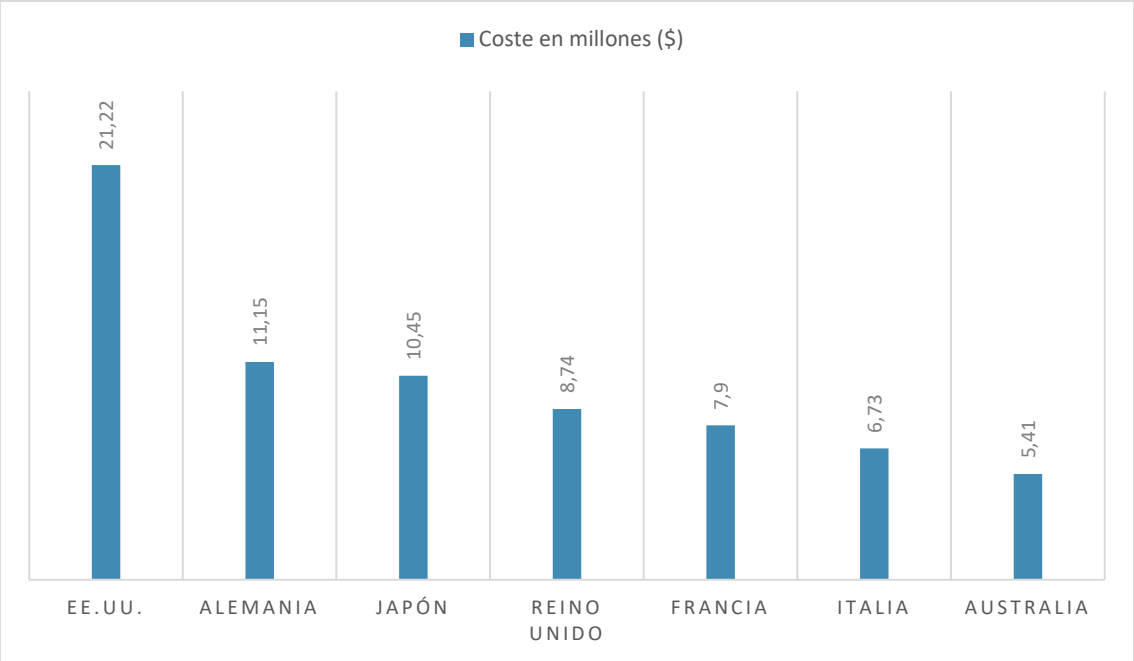

Figura 1 Coste de ataques informáticos por país en 2017 (millones de $)

Además, ninguna industria ni sector empresarial están libres de sufrir un ataque informático, ya que éstos no solo se producen en empresas tecnológicas. De hecho, según un estudio llevado a cabo por la empresa Marsh & McLennan Companies [4], el porcentaje de empresas de cada sector principal que había sufrido un ataque durante el año 2017 fue:

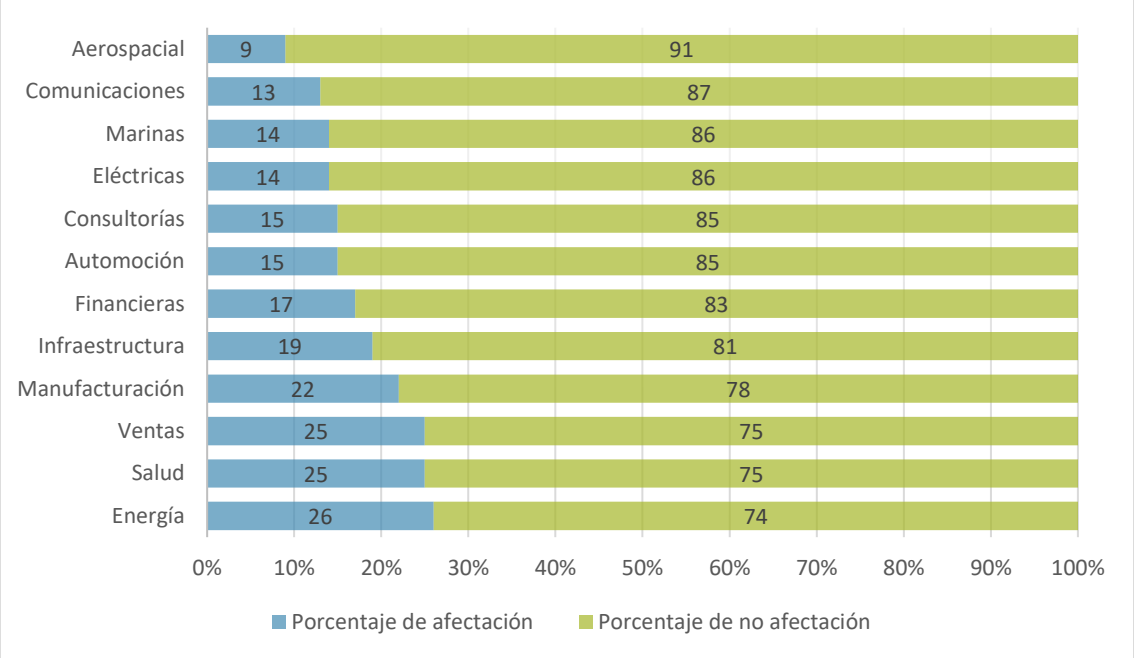

Figura 2 Industrias afectadas por ataques informáticos en 2017

Todos los ciberataques y brechas de seguridad que se producen diariamente no tienen la misma metodología, el mismo origen, ni el mismo resultado. En el mismo estudio realizado por Accenture y Ponemon Institute [2], se detalla el porcentaje de empresas que han sufrido un ataque informático de un tipo determinado en 2017, y como ha crecido desde 2016:

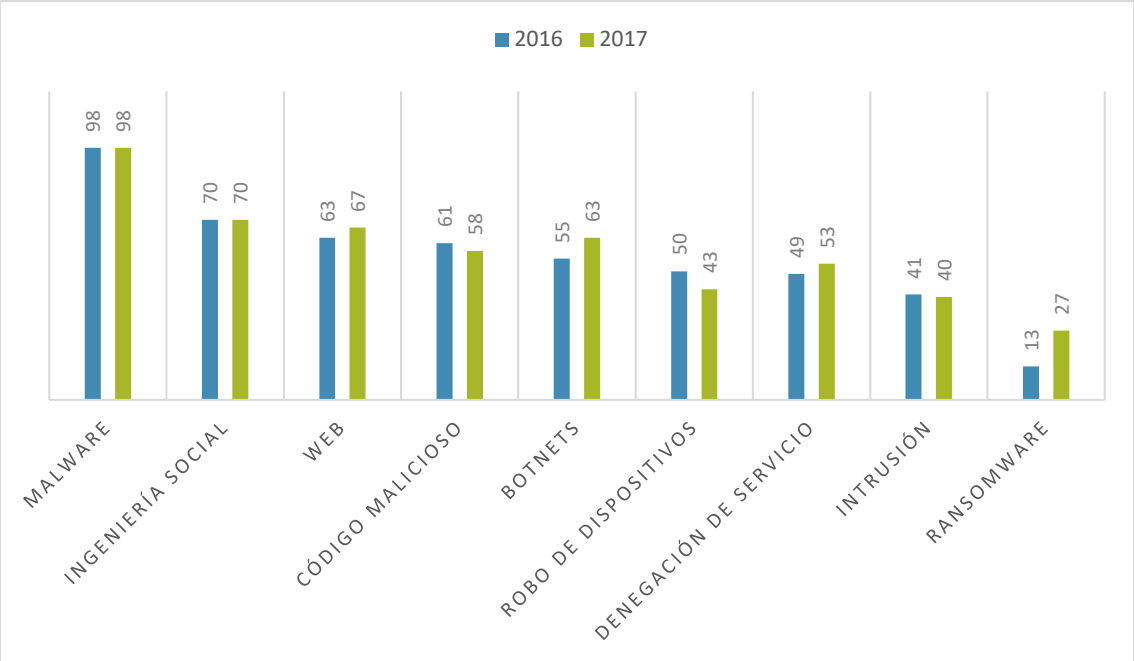

Figura 3 Técnicas más frecuentes en ataques informáticos en 2017

Además, hay que tener en cuenta un factor, que es el del conocimiento o desconocimiento del ataque informático que se está produciendo. Muchas empresas sufren ataques informáticos a través de malware o técnicas que son conocidas por que se han producido con anterioridad, pero existe la posibilidad de que el ataque informático sea desconocido, debido a que se explota una vulnerabilidad desconocida o se utiliza una técnica jamás vista con anterioridad. A estos ataques informáticos nuevos se les denomina zero-days, y suponen un gran peligro para las empresas, debido a la dificultad de protección contra algo que se desconoce. Según el informe trimestral de la empresa Watchguard [5], un 30% de los ataques analizados por los antivirus no son detectados y clasificados como zero-days. Un porcentaje del 30% entre el número de ataques informáticos que se producen diariamente es un porcentaje altísimo, por lo que el papel que juegan la detección, investigación y reporte de ataques zero-days es cada vez más y más importante.

3 Descripción de la situación actual

La Diputación Provincial de Cádiz dispone de una Red Corporativa Provincial de telecomunicaciones, suministrada por Telefónica, que proporciona servicio de datos y voz a todas las dependencias de la Diputación, sus organismos autónomos y entidades locales a los que da servicio.

3.1 Arquitectura global

La arquitectura global de la Red Corporativa Provincial de telecomunicaciones de la Diputación Provincial de Cádiz es la siguiente:

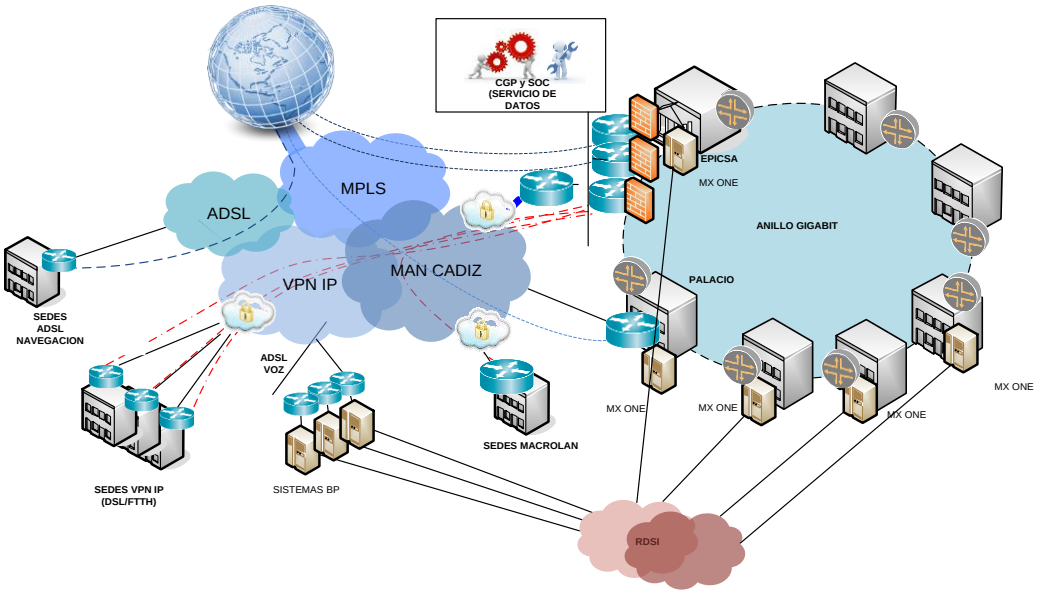

Figura 4 Red de la Diputación Provincial de Cádiz

Para la red de datos metropolitana se ha desplegado un anillo de fibra óptica con ancho de banda de 1 Gbps. Dicho anillo de fibra óptica interconecta todas las sedes de la Diputación Provincial de Cádiz con situación en la propia ciudad de Cádiz.

Para conectar las sedes de la Diputación Provincial de Cádiz externas al anillo metropolitano se ha desplegado una Red Privada Virtual (VPN) que, mediante la tecnología VPN IP de Telefónica, se conectarán dichas sedes remotas a la sede principal ubicada en EPICSA.

El acceso a Internet de la red de la Diputación Provincial de Cádiz se separa en dos tipos de accesos:

- **Internet para publicar.** Este acceso se reserva para el acceso a aplicaciones y servicios propios de la Diputación Provincial de Cádiz. Dicho acceso se soporta sobre un acceso MacroLan de fibra óptica situado en EPICSA.
- **Internet para navegar.** Este acceso se utiliza para el resto de las conexiones a Internet. Para las sedes del anillo de red metropolitano se ofrecen dos caudales de navegación sobre MacroLan de fibra óptica en las sedes de EPICSA y Palacio Provincial. Para las sedes remotas se provee acceso a Internet a través de tecnología DSL.

En lo referente a la seguridad de la red de la Diputación Provincial de Cádiz, se dispone de:

- **Protección de los servicios de publicación.** En la sede de EPICSA, se despliega un clúster de dos Fortigate 600C que realizan la función de cortafuegos.
- **Protección del servicio de navegación para la red metropolitana.** En la sede de EPICSA, se despliega un clúster de dos Fortigate 880C Bundle, que realizan las funciones de cortafuegos, antivirus web y filtrados de URL.
- **Protección del servicio de navegación para oficinas remotas.** Se dispone de un servicio Cloud de navegación segura en modalidad esencial para 480 usuarios nominales con administración delegada en el cliente.

3.2 Modularización de la red

Según el planteamiento teórico que Cisco realiza sobre el diseño funcional de una red [6], esta se puede dividir en diferentes áreas funcionales que, a su vez, se dividen en diferentes módulos. Si aplicamos la aproximación del diseño funcional de una red al diseño de la red de la Diputación de Cádiz, las áreas funcionales quedan distribuidas de la siguiente manera:

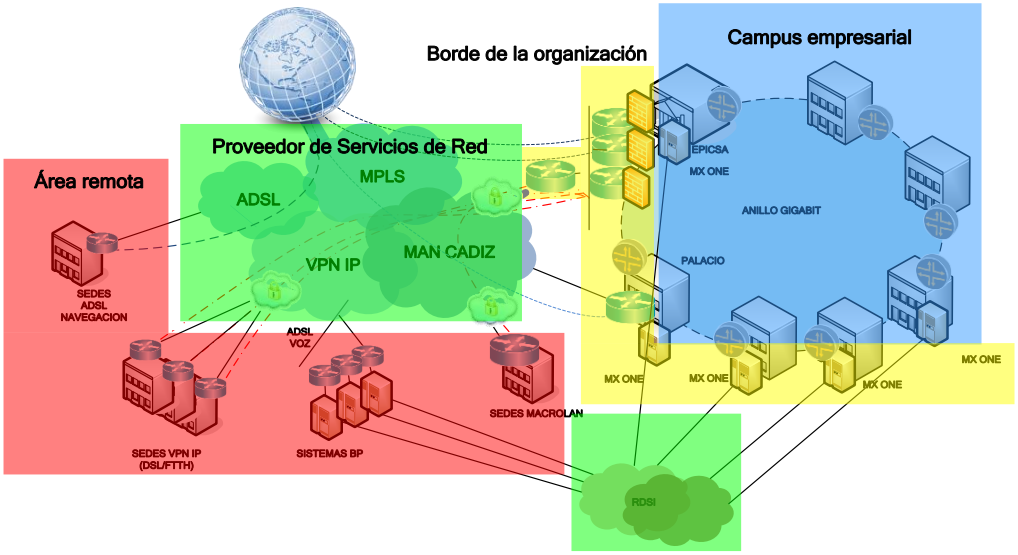

Figura 5 Modularización de la red de la Diputación de Cádiz

3.2.1 Campus empresarial

El área del campus empresarial es la sede principal de una organización y, a su vez, se divide en el módulo de infraestructura de red y en el módulo de centro de procesamiento de datos y granja de servidores.

En lo referente a la infraestructura de red, la Diputación Provincial de Cádiz dispone de una red metropolitana en anillo que interconecta las diferentes sedes de la diputación de Cádiz con residencia en la ciudad de Cádiz. Cada una de las sedes de las que dispone la Diputación Provincial de Cádiz corresponde a un nodo de la red metropolitana en anillo, salvo algunas que, por cercanía a otras sedes, se disponen como conexiones directas a otras sedes y no pertenecen al anillo propiamente dicho.

La Diputación de Cádiz posee un centro de procesamiento de datos (CPD) con sede en EPICSA, que se enmarca en el módulo de centro de procesamiento de datos y granja de servidores.

3.2.2 Borde de la organización

La infraestructura de borde de la organización agrupa la conectividad de varios dispositivos externos al campus de la organización y enruta el tráfico hacia la capa de núcleo de la infraestructura de red interna. Los módulos pertenecientes al área de borde de la organización ofrecen funcionalidades de seguridad que securizan los recursos de la organización cuando se producen conexiones con redes públicas y/o Internet.

Se ha desplegado una red desmilitarizada (DMZ) para ofrecer servicios corporativos a la red pública. Dicha red DMZ se conecta a la red pública a través de un clúster de firewalls Fortigate 600C, que a su vez se conecta a un clúster de firewalls Checkpoint que se conecta directamente al ISP. El acceso público desde Internet a la DMZ se realiza a través de estos dos clústeres de firewalls.

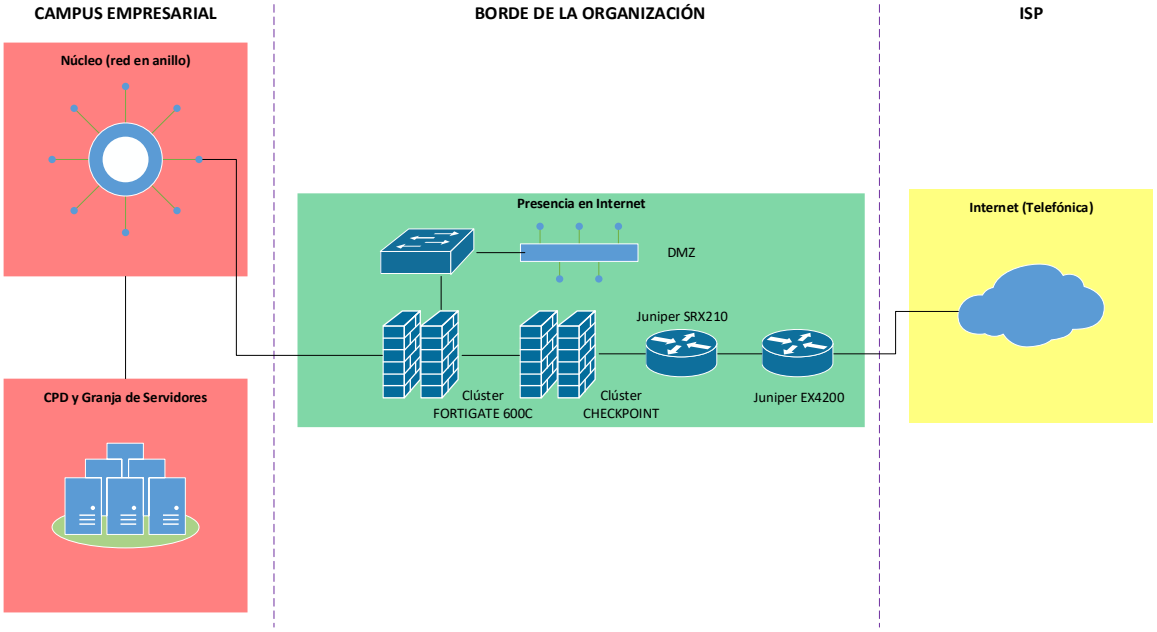

Figura 6 Zona DMZ en la red de la Diputación de Cádiz

3.2.3 Proveedor de Servicios de Red

En la red de la Diputación provincial de Cádiz existen conexiones a diferentes ISP y diferentes servicios de estos, según el caudal de tráfico correspondiente. Todos los EDC desplegados en la red de la Diputación de Cádiz para conectarse a los diferentes ISP son gestionados por dichos ISP.

La empresa EPICSA, para ofrecer servicios de datos y aplicaciones al Ayuntamiento de Puerto Real, tiene contratado servicio de red con el proveedor ONO. Se tiene contratado con el Proveedor ONO una línea con ancho de banda de 10 Mb/s simétricos. La red del proveedor finaliza en el EDC de segundo nivel que se conecta con el EDC de primer nivel. Es el propio proveedor ONO el que gestiona la línea de conexión con el Ayuntamiento de Puerto Real.

La empresa EPICSA, para ofrecer servicios de datos y aplicaciones a la Diputación de Cádiz (sedes remotas, servicios provinciales de recaudación, ayuntamientos con menos de 20.000 habitantes, etc.), tiene contratado servicio de red con el proveedor Telefónica. Se tiene contratado con el Proveedor Telefónica una línea con ancho de banda de 10 Mb/s simétricos. La red del proveedor finaliza en los diferentes EDCs de segundo nivel, desplegados en EPICSA, que se conectan con los dos EDCs de primer nivel. Es el propio proveedor Telefónica el que gestiona las líneas de navegación a Internet, de presencia corporativa en la web, de VPN sitio a sitio, de conexión con la sede SS.CC de la Diputación de Cádiz y de las VPNs de acceso remoto.

Para la presencia corporativa en la web, se le ha asignado a EPICSA tres bloques de direcciones IP públicas:

- 213.0.62.64/29
- 213.0.60.32/29
- 194.179.87.16/29

Cabe destacar que la empresa EPICSA no dispone de sistema autónomo propio.

3.2.4 Área remota

EPICSA ofrece conexión a la red NEREA de la Junta de Andalucía al Ayuntamiento de Puerto Real a través de los servicios de SigADI de ONO. Se ha desplegado un router Cisco C1700-SV8Y7-M que establece la conexión con la red de la Diputación de Cádiz.

La empresa EPICSA ofrece conexión a la red de la Diputación de Cádiz mediante el servicio VPN IP de Telefónica a Servicios Provinciales de Recaudación, sedes remotas de la Diputación de Cádiz y a los ayuntamientos de las ciudades de la Provincia de Cádiz con menos de 20.000 habitantes. La conexión garantiza un ancho de banda mínimo de 2 Mbps. de bajada y 512 Kbps de subida.

Para la conexión a la red de la Diputación de Cádiz de la sede de Servicios Sociales y Comunitarios, se ha desplegado un router Cisco 887-SEC-K9. El acceso a Internet de la sede si es competencia de EPICSA por lo que el servicio lo presta Telefónica a través del convenio existente con la Diputación de Cádiz. La elección y gestión del equipo desplegado para el acceso a Internet son competencias de Telefónica.

Para la conexión a la red de la Diputación de Cádiz, un teletrabajador puede utilizar cualquier conexión a Internet. Simplemente tiene que conectarse al servidor VPN de EPICSA a través de la IP pública: 195.55.52.245 y utilizar el certificado expedido por la misma EPICSA.

4 Normas y referencias

4.1 Disposiciones legales y normas aplicadas

- Norma UNE 157801:2007**.** Criterios generales para la elaboración de proyectos de sistemas de información.

4.2 Bibliografía

[1] CCN-CERT, «CCN-CERT IA-16/17 Ciberamenzas y Tendencias. Edición 2017». [En línea]. Disponible en: https://www.ccn-cert.cni.es/en/reports/public/2224-ccn-cert-ia-16-17-ciberamenzas-y-tendencias-edicion-2017. [Accedido: 02-ene-2018].

[2] Accenture, «2017 Cost of Cyber Crime Study | Accenture». [En línea]. Disponible en: https://www.accenture.com/us-en/insight-cost-of-cybercrime-2017. [Accedido: 31-ene-2018].

[3] Accenture y Ponemon Institute, «Cyber crime: average company loss in selected countries 2017 | Statistic», *Statista*. [En línea]. Disponible en: https://www.statista.com/statistics/293274/average-cyber-crime-costs-to-companies-in-selected-countries/. [Accedido: 31-ene-2018].

[4] Marsh & McLennan Companies, «MMC Cyber Handbook 2018: Perspectives on the next wave of cyber». [En línea]. Disponible en:

https://www.marsh.com/us/insights/research/mmc-cyber-handbook-2018.html. [Accedido: 31-ene-2018].
[5] Watchguard, «Internet Security Report - Q3 2017», 27-mar-2017. [En línea]. Disponible en: https://www.watchguard.com/wgrd-resource-center/security-report. [Accedido: 31-ene-2018].
[6] Sean Wilkins, «Designing for Cisco Internetwork Solutions (DESGN) Foundation Learning Guide: (CCDA DESGN 640–864), Third Edition [Book]». [En línea]. Disponible en: https://www.safaribooksonline.com/library/view/designing-for-cisco/9780132582407/. [Accedido: 31-ene-2018].
[7] SC Media US, «Intrusion Prevention Systems Reviews». [En línea]. Disponible en: https://www.scmagazine.com/intrusion-prevention-systems/products/6516/0/. [Accedido: 01-jun-2018].
[8] EmergingThreats, «WebHome - Main - EmergingThreats». [En línea]. Disponible en: http://doc.emergingthreats.net/. [Accedido: 04-jun-2018].
[9] «Oinkmaster». [En línea]. Disponible en: http://oinkmaster.sourceforge.net/. [Accedido: 04-jun-2018].
[10] CVE, «Apache Http Server version 2.4.18 : Security vulnerabilities». [En línea]. Disponible en: https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-199589/Apache-Http-Server-2.4.18.html. [Accedido: 07-jun-2018].
[11] «Openbsd Openssh version 7.2 : Security vulnerabilities». [En línea]. Disponible en: https://www.cvedetails.com/vulnerability-list/vendor_id-97/product_id-585/version_id-194112/Openbsd-Openssh-7.2.html. [Accedido: 07-jun-2018].
[12] CVE, «Microsoft IIS version 6.0 : Security vulnerabilities». [En línea]. Disponible en: https://www.cvedetails.com/vulnerability-list/vendor_id-26/product_id-3436/version_id-13492/Microsoft-IIS-6.0.html. [Accedido: 07-jun-2018].
[13] Pentest-Tools, «Online Penetration Testing and Ethical Hacking Tools», *Pentest-Tools.com*. [En línea]. Disponible en: https://pentest-tools.com/home. [Accedido: 07-jun-2018].
[14] Palo Alto Networks, «How to Break the Cyber Attack Lifecycle - Palo Alto Networks». [En línea]. Disponible en: https://www.paloaltonetworks.com/cyberpedia/how-to-break-the-cyber-attack-lifecycle. [Accedido: 02-ene-2018].
[15] Niels Provos, «A Virtual Honeypot Framework». [En línea]. Disponible en: https://deepblue.lib.umich.edu/handle/2027.42/107882. [Accedido: 02-ene-2018].
[16] Fahim Huda Abbasi, «Building and Deploying a GenIII Virtual Honeynet». [En línea]. Disponible en: http://seat.massey.ac.nz/projects/honeynet/honeynet.htm. [Accedido: 02-ene-2018].
[17] Honeynet Project, «Know Your Enemy: Honeynets». [En línea]. Disponible en: http://old.honeynet.org/papers/honeynet/. [Accedido: 02-ene-2018].
[18] Honeynet Project, «Know Your Enemy: Virtual Honeynets». [En línea]. Disponible en: http://old.honeynet.org/papers/virtual/. [Accedido: 02-ene-2018].

4.3 Métodos, herramientas, modelos, métricas y prototipos

- **GanttProject.** Editor de diagramas de Gantt utilizado para la realización del diagrama de planificación del proyecto**.**
- **Microsoft Visio 2016**. Editor de gráficos utilizado para la edición de esquemas visuales.
- **Microsoft Word 2016**. Editor de texto utilizado para la elaboración de la memoria del proyecto.

5 Definiciones y abreviaturas

- **ADSL**: Asymmetric Digital Subscriber Line.
- **ASA**: Advanced Security Appliance.
- **CPD**: Centro de procesamiento de datos.
- **CPU**: Central Processing Unit.
- **CVE**. Vulnerabilidades y Exposiciones Comunes (En inglés, *Common Vulnerabilities and Exposures*).
- **DDoS**. Denegación de Servicio Distribuida.
- **DMZ**: Red desmilitarizada.
- **DNS**. Sistema de Nombres de Dominio.
- **DNS**: Sistema de resolución de nombres.
- **DoS**. Denegación de Servicio.
- **EDC**: Equipo en domicilio del cliente.
- **FTP**: File Transfer Protocol.
- **HDD**: Hard Drive Disk.
- **HIDS**: Sistema de detección de intrusiones en hosts.
- **HTTP**: Protocolo de transferencia de hipertexto.
- **HTTPS**: Protocolo seguro de transferencia de hipertexto.
- **IDS**. Sistema de detección de intrusiones.
- **IP:** Protocolo de internet.
- **IPS:** Sistema de prevención de intrusiones
- **ISP**: Proveedor de Servicios de Internet.
- **JSON**: JavaScript Object Notation
- **Kbps**: Kilobits por segundo.
- **Mbps**: Megabits por segundo.
- **MBps**: Megabytes por segundo.
- **NAT**: Network Address Translation.
- **NFQ**: Netfilter Qeue.
- **NIPS**: Sistema de prevención de intrusiones en red.
- **OISF**: Open Information Security Foundation
- **PF_RING**: High-speed packet capture, filtering and analysis
- **RAM**: Random Access Memory.
- **SMTP**: Simple Mail Transfer Protocol.
- **SSH**: Secure Shell.
- **URL**: Lozalizador de recursos uniforme.
- **VPN**: Red virtual privada.

6 Requisitos iniciales

Los objetivos del proyecto de despliegue de una honeynet virtual para la detección y el estudio de ataques informáticos son los siguientes:

- **R-01: Desplegar una honeynet que no sobrecargue la infraestructura existente**. La arquitectura de la honeynet desplegada no debe sobrecargar la infraestructura de red en producción.
- **R-02: Desplegar una honeynet escalable**. La arquitectura de la honeynet debe permitir la rápida y fácil escalabilidad de los servicios desplegados.
- **R-03: Implementar control y limitación de conexiones**. La honeynet debe controlar y limitar todas las sesiones establecidas con honeypots de la red vulnerable.
- **R-04: Implementar prevención de intrusiones en red**. La honeynet debe ser capaz de alertar y bloquear ante la detección de tráfico de red sospechoso.
- **R-05: Implementar detección de intrusiones en honeypots**. La honeynet debe ser capaz de alertar ante la detección de software malicioso en los honeypots.
- **R-06: Implementar control de integridad en honeypots**. La honeynet debe ser capaz de alertar ante el cambio de ficheros o software existentes en el sistema, así como la aparición de nuevos ficheros o software.
- **R-07: Implementar registro de inicios de sesión**. La honeynet debe capturar todos los registros de inicio de sesión en los diferentes honeypots.
- **R-08: Implementar registro de comandos**. La honeynet debe capturar todos los registros de comandos emitidos en los diferentes honeypots.
- **R-09: Implementar captura de tráfico**. La honeynet debe capturar todo el tráfico que la atraviese
- **R-10: Recoger estadísticas de uso y sesiones de la red**. La honeynet debe generar estadísticas de uso de la red con información de las diferentes sesiones establecidas.
- **R-11: Recoger estadísticas de uso de recursos de la honeynet**. La honeynet debe generar estadísticas de uso de los recursos de los honeypots y el honeywall.
- **R-12: Implementar cifrado en la colección de datos**. Todo intercambio de información entre el honeywall y los honeypots debe ser a través de canales cifrados.
- **R-13: Implementar un sistema de análisis de información**. La honeynet debe provisionar de un sistema de análisis de información residente en el honeywall.
- **R-14: Configurar una conexión cifrada con el honeywall**. Toda conexión que se establezca con el honeywall ya sea para análisis de datos o administración de la honeynet, debe ser cifrada.

7 Alcance

Este proyecto se aplica al diseño y despliegue de una honeynet virtual para la detección y análisis de ataques informáticos en un entorno de investigación.

Este proyecto incluye:

- Especificación de requisitos de la honeynet virtual
- Introducción teórica a los ataques informáticos
- Estudio teórico de los honeypots.
- Estudio teórico de las honeynets.
- Establecimiento de mediciones de recursos para la consecución del proyecto.
- Elaboración de un presupuesto para la consecución del proyecto.
- Estudio de alternativas y viabilidad de las diferentes alternativas hardware y software para la realización del proyecto.
- Descripción de las soluciones adoptadas.
- Anexo de información sobre la configuración del software utilizado.

8 Estudio de alternativas y viabilidad

En este apartado se estudiarán las posibles alternativas a considerar en la consecución de este proyecto. Las siguientes alternativas se desarrollarán en los apartados sucesivos.

Concepto	Alternativas a estudiar
Arquitectura de la honeynet	Generación de la arquitectura
Despliegue de la honeynet	Virtualización de servicios
Hardware de despliegue	CPU, RAM, HDD, Interfaces de red
Software del honeywall	Control, captura, colección y análisis de datos
Software de los honeypots	Interacción, servicios vulnerables

Tabla 1 Resumen de estudio de alternativas y viabilidad

8.1 Arquitectura de la honeynet

En la sección del estudio teórico del presente proyecto se establecen tres diferentes arquitecturas de despliegue posibles para una honeynet: primera, segunda o tercera generación. A la hora de decidir que generación seguir en el despliegue de una honeynet, debemos tener en cuenta algunas consideraciones.

El despliegue de una honeynet de primera generación, donde toda la honeynet se separa de la red en producción mediante el cortafuegos frontera de la red, supone el despliegue y configuración de una nueva red estructurada: puertos del cortafuegos, acceso a internet, limitación de conexiones, configuración de IDS/IPS en el cortafuegos, etc. Posiblemente, una empresa pequeña con pocos recursos no pueda permitirse el despliegue de una nueva red estructurada para la honeynet, porque a lo mejor ni si quiera tenían un cortafuegos anteriormente. Para una empresa más grande, el coste si puede ser admisible, aunque la integración de una nueva red estructurada con las demás redes en producción supone nuevos retos para la configuración del cortafuegos frontera, como el aislamiento de todas las redes con la honeynet, la asignación de bloques de direcciones, utilización de más interfaces de red, configuración de IDS/IPS y/o captura de tráfico, etc. Todo esto dependerá del tamaño de la empresa, la complejidad de sus redes en producción y del poder adquisitivo de cada una.

Al desplegar una honeynet de segunda generación, la separación de las tareas de control de la honeynet al honeywall, dejando al cortafuegos frontera únicamente como el punto de separación de la honeynet con las demás redes en producción, hace que las honeynets de segunda generación sean más sencillas de desplegar en redes complejas con cortafuegos instalados previamente. El coste de despliegue sería el mismo en ambas generaciones, pero la configuración del cortafuegos frontera en una honeynet de segunda generación es más sencilla, al solo necesitar de habilitar el acceso a Internet en la honeynet y aislar las demás redes de la honeynet.

Las honeynets de tercera generación son la opción más favorable para las pequeñas empresas al no suponer ningún coste en el despliegue de nuevas redes estructuradas. Toda la honeynet estará contenida dentro de la red en producción. En grandes empresas, donde existe una red en producción compleja con diferentes subredes, el despliegue de una honeynet, o incluso más, en diferentes zonas de la red interna supone un gran reto de configuración, debido a

que el aislamiento de la red honeynet con los equipos en producción es muy delicado, y debe ser configurado al detalle.

8.2 Despliegue de la honeynet

En el estudio del despliegue de una honeynet, se debe decidir si se virtualizarán los honeypots y el honeywall (honeynets independientes), si se virtualizarán solo los honeypots (honeynet híbrida), o si, por el contrario, no se virtualizará ningún servicio (honeynet clásica).

El despliegue de una honeynet clásica, sin ningún servicio virtualizado, conlleva los gastos usuales de adquisición y configuración, tanto de software como de hardware, costes que pueden ser asumibles para una empresa grande, pero que podrían no serlo para una empresa pequeña. Por eso, se deben tener en cuenta otras alternativas de despliegue que integran servicios virtualizados.

Las honeynets independientes son las honeynets que menos coste suponen a la hora de adquirir hardware, ya que toda la honeynet se virtualiza dentro de una misma plataforma física. Aun así, si se desea desplegar una honeynet compleja, de segunda o tercera generación, que soporte mucho tráfico y análisis de datos, el hardware para la virtualización de toda la infraestructura debería tener unos requerimientos hardware tan altos que se debería considerar la adquisición de diferentes plataformas físicas de virtualización, para separar todo el trabajo de computación.

A la hora de desplegar una honeynet que tenga que soportar un alto tráfico de red, así como un gran análisis de datos, las honeynets virtuales son la mejor opción. Un honeywall independiente asegura una completa dedicación al control, captura y análisis de datos de una honeynet, mientras que otro servidor de virtualización contendrá todos los honeypots virtualizados que componen la honeynet. El coste de despliegue es superior al de una honeynet independiente, pero sigue siendo menor que el de una honeynet clásica sin servicios virtualizados.

8.2.1 Software de virtualización

A la hora de elegir un software de virtualización disponemos de muchas posibilidades en el mercado actual, aunque las dos plataformas de virtualización más conocidas son VMware, VirtualBox y Proxmox.

VMWare es la plataforma líder en virtualización en el mercado actual. Sus posibilidades abarcan desde la creación y gestión de máquinas virtuales locales, con su versión gratuita VMware Player, hasta la gestión de clústeres de servidores de virtualización para su uso como nube de máquinas virtuales, con la versión de pago para empresas VMWare ESXi. VirtualBox es una plataforma de virtualización de código abierto de Oracle disponible para la mayoría de los sistemas operativos disponibles para la creación y gestión de máquinas virtuales locales. Por último, La empresa Proxmox, ofrece Proxmox VE, una plataforma de creación y gestión de servidores de virtualización de código abierto que ofrece grandes herramientas para la gestión de máquinas virtuales en la nube: alta disponibilidad, almacenamiento, gestión remota (SSH/Web), etc.

La elección de cada una de estas alternativas de plataformas de virtualización dependerá del contexto del estudio y del presupuesto disponible, siendo VirtualBox la opción perfecta para

usuarios nuevos en virtualización o cuyas necesidades abarcan solo algún tipo de prueba rápida. VMWare es la solución más desplegada para entornos empresariales donde, pudiendo asumir el coste de la licencia de VMWare EXSi, se configuran máquinas virtuales en la nube. Proxmox es la solución de plataforma de virtualización en la nube perfecta para investigación y aprendizaje de nuevas herramientas de virtualización, aunque, con un buen mantenimiento y control, puede operar perfectamente en entornos empresariales.

8.3 Hardware de despliegue

Dependiendo de la arquitectura de la honeynet que se vaya a desplegar, así como de la virtualización de diferentes servicios que queramos configurar, el hardware necesario para implantar una honeynet puede variar, en lo referente a CPU, memoria RAM, capacidad de HDD y número de interfaces de red.

Para honeynets de primera generación, se deberá adquirir hardware cuyos requerimientos sean directamente proporcionales a la cantidad de tráfico que vayan a soportar los honeypots y de la complejidad de los servicios instalados, es decir, para honeypots de alta interacción que soporten gran cantidad de conexiones, los requerimientos hardware serán altos (CPU y RAM) con poco uso de HDD y una sola interfaz de red. Mientras que para honeypots de baja interacción con poca carga de direcciones no será necesario un gran despliegue de CPU y RAM.

Para honeynets de segunda y tercera interacción hay que tener en cuenta que, en adición a lo expresado anteriormente, debemos añadir el honeywall, que deberá soportar todo el tráfico de la honeynet y deberá proveer de todas las herramientas de control, captura y análisis de datos de la honeynet. Un honeywall debería tener un buen rendimiento de CPU y de RAM, así como bastante almacenamiento masivo (HDD) y, como mínimo, tres interfaces de red, cuyo ancho de banda dependerá de las características de la red de la empresa. Como se dijo anteriormente, estos requerimientos variarán según las características específicas de la red en producción sobre la que se despliegue la honeynet.

Si se desea desplegar una honeynet virtualizada, ya sea independiente o híbridas, hay que tener en cuenta que el hardware sobre el que se vayan a virtualizar servicios debe poder soportar la virtualización de software y toda la carga computacional que suponga la virtualización de honeypots, o de honeypots y honeywall. Si se despliega una honeynet independiente, el servidor de virtualización debe tener grandes recursos de computación (CPU, HDD y RAM) aunque solo necesita una interfaz de red física (el resto se virtualizan). Para el despliegue de una honeynet híbrida, el servidor de virtualización necesitará menos recursos computacionales al solo soportar la carga de los honeypots.

8.4 Software del honeywall

8.4.1 Control de datos

Tal y como se especifica en el apartado teórico, uno de los procesos necesarios que deben existir en el honeywall de una honeynet de segunda o tercera generación es el control de datos. Para cumplir con el control de datos, el honeywall debe ser capaz de controlar qué puede hacer un atacante dentro de la honeynet o no, ya sea lanzando ataques externos

dirigidos a los honeypots o ataques hacia Internet con origen en los honeypots, previo compromiso de estos por parte del atacante.

El primer mecanismo de control que se debe analizar es el de limitación de conexiones y ancho de banda, es decir, permitir a un atacante el número de conexiones y el uso de ancho de banda justo y necesario para poder atacar a los honeypots, pero lo suficiente como para provocar una denegación de servicio, tanto hacia los honeypots o hacia Internet desde los honeypots. Para implementar la limitación de conexiones se dispone de varias alternativas:

- Implementar dicho control en el cortafuegos frontera de la honeynet. Con esta alternativa, propia de una honeynet de primera generación, el cortafuegos perimetral de la honeynet controlaría y, consecuentemente y cuando sea necesario, limitaría las conexiones de los posibles atacantes.
- Implementar el control en el honeywall. Esta alternativa, propia de honeynets de segunda y tercera generación, delega la responsabilidad del control de conexiones al servidor puente encargado del acceso a la honeynet.

En el caso de implementar el control de datos en el honeywall, se disponen de varias alternativas, según el sistema operativo desplegado en el servidor de control:

- GNU/Linux: los sistemas GNU/Linux disponen de manera nativa del software IPTables, un cortafuegos cuyo funcionamiento se implementa mediante Netfilter, un subsistema de procesamiento de paquetes en red del kernel de Linux. La posibilidad de uso de IPTables es beneficiosa ya que, debido a su integración con el núcleo de Linux, su rendimiento está muy optimizado e integrado con todos los demás procesos residentes en el núcleo. La única desventaja que propone IPTables es su dificultad de administración, ya que todo el proceso de mantenimiento y actualización se realiza a través de línea de comandos
- Cortafuegos como sistema operativo: también existe la opción de instalar en el honeywall un sistema operativo cuya totalidad se base en ser utilizado como cortafuegos. Actualmente existen muchas posibilidades: ClearOS, IPCop, Verdict, OPNsense, etc., todas ellas con el mismo funcionamiento básico de cortafuegos. Las ventajas que presentan estos sistemas operativos dedicados son su sencillez de administración, ya que la mayoría disponen de interfaces web propias y su buen rendimiento, ya que todo el software del sistema operativo está dedicado a la función de cortafuegos. La desventaja principal que presentan estos sistemas operativos dedicados son la imposibilidad de instalación de software con diferente propósito, lo que supondría que el honeywall solo se podría dedicar a ser un cortafuegos, y no podría incorporar otros tipos de procesos de la honeynet.

El segundo mecanismo de control de datos que se debe analizar en una honeynet es, no sólo limitar las conexiones y ancho de banda de los atacantes sino alertar e incluso bloquear aquellos tipos de ataques que ya sean conocidos con anterioridad. Una honeynet que nos alerte de un tráfico que ya ha sido identificado como malicioso introduce un factor de ruido a la hora de tratar de identificar nuevas técnicas de ataque y tráfico malicioso desconocido, que es el objetivo principal del despliegue de una honeynet, la investigación de nuevas

técnicas de ataque. Para ello, el mecanismo más eficiente que se puede desplegar en una honeynet es un IPS, sistema de prevención de intrusiones, que bloquee el tráfico malicioso conocido y deje pasar el tráfico desconocido susceptible de ser malicioso para su posterior investigación. Para desplegar un IPS en la honeynet, disponemos de dos alternativas, según la generación de esta:

- Primera generación: el IPS sería instalado en el cortafuegos frontera de la propia honeynet. Esto tendría la desventaja de que el despliegue del IPS estaría sujeto a la compatibilidad de este con el cortafuegos, o incluso, limitado a la solución IPS que el fabricante del cortafuegos proponga, lo que dificultaría su integración con otras herramientas de colección y análisis de datos.
- Segunda y tercera generación: el IPS sería instalado en el servidor que actúa como honeywall. La ventaja principal de esta opción son las diferentes posibilidades en la elección de un IPS para sistemas operativos basados en GNU/Linux. La desventaja de esta opción es que la elección del IPS está sujeta a su integración con los demás mecanismos de captura, colección y análisis de datos de la honeynet, así como de todo el software que los implementan.

En el caso de seleccionar la segunda opción, disponemos de varias posibilidades para desplegar un IPS, tanto soluciones privativas como de código abierto:

- Privativas: las soluciones privativas de IPS son soluciones que tienen como principal ventaja el apoyo técnico que ofrecen los vendedores de estas. La desventaja principal que presentan son su frecuente alto coste económico. Algunas soluciones privativas de IPS son [7]:
    - Metaflows Security System:
        - Vendedor: Metaflows.
        - Tipo: basado en la nube.
        - Precio: desde 1.089€ al año.
    - Sourcefire Next-Generation IPS:
        - Vendedor: Sourcefire.
        - Tipo: basado en *appliances*.
        - Precio: 8.995€.
    - Nitroguard IPS 4245:
        - Vendedor: NitroSecurity.
        - Tipo: software instalado en un servidor.
        - Precio: desde 6.495€ para versiones de bajo rendimiento hasta 54.495€ para versiones de mayor rendimiento.
- Código abierto: las soluciones de código abierto ofrecen el despliegue de un IPS a coste económico nulo, mientras que presentan la desventaja de la falta de apoyo técnico por parte de una empresa. Las dos principales soluciones de código abierto IPS son:
    - Snort, desarrollado por SourceFire es el estándar de IDS/IPS de código abierto actualmente, con características como:
        - Facilidad de instalación.
        - Buena documentación.

        - Exportación de datos de alerta y bloqueo a diferentes formatos.
        - No soporta ejecución multi-hilo.
    - Suricata, desarrollado por Open Information Security Foundation (OISF), nació como una versión alternativa de Snort que, a parte de la funcionalidad básica de Snort, añade:
        - Ejecución multihilo.
        - Soporte IPv6 completo.
        - Exportación de datos en formato JSON.
        - Aceleración de captura de paquetes mediante la tecnología PF_RING.

8.4.2 Captura de datos

Tal y como se describe en el apartado teórico, el proceso de captura de datos de una honeynet involucra todos los mecanismos de esta que provean de la mayor captura de datos posible para que, la investigación de los ataques informáticos que se produzcan esté respaldada con la máxima cantidad de datos posible.

A la hora de implementar todos los mecanismos de captura de datos se debe analizar, primero, que tipo de datos se desean capturar y posteriormente, que alternativas existen para implementar la captura de cada uno de los tipos de datos seleccionados.

En una honeynet es interesante capturar la máxima cantidad de datos posibles y de la mayor variedad posible en su tipología. Normalmente, en una honeynet se deben capturar los siguientes datos:

- Capturas del tráfico que atraviesa la honeynet, en ambos sentidos.
- Datos generados por los mecanismos involucrados en el proceso de control de datos.
- Estadísticas de todas las conexiones que se establecen en la honeynet.
- Estadísticas de uso del hardware del honeywall.
- Estadísticas de uso del hardware de los honeypots.
- Estadísticas de uso del servidor de virtualización de honeypots (si procede).
- Datos de intentos de inicios de sesión en los honeypots, tanto fallidos como correctos.
- Datos relacionados con los diferentes servicios que se desplieguen en los honeypots, por ejemplo:
    - Servicio web: datos de conexión como User Agents, URL de conexión, parámetros, carga de datos de la petición, etc.
    - Servicio FTP: registro de intentos de inicio de sesión, registro de transacción de ficheros, etc.
    - Servicio SMTP: registro de direcciones de correo involucradas, captura de los contenidos del correo, incluyendo adjuntos (si existen).

8.4.3 Colección de datos

Una vez que todos los mecanismos de la honeynet que implementan el proceso de captura de datos, se debe implementar el proceso de colección de datos que, tal y como se especifica

en el apartado teórico, se encarga de recolectar todos los datos generados en una honeynet para su posterior análisis por parte del administrador de ésta.

Para desplegar un mecanismo de colección de datos eficiente en la honeynet, se debe tener en cuenta que dicho mecanismo debe soportar una gran variedad de datos de diferente tipología, pues los datos generados en una honeynet son muy heterogéneos. En el caso de que la honeynet genere muchos datos en un espacio corto de tiempo, el mecanismo de colección de datos debe ser capaz de soportar toda la recolección de esos datos sin producir retrasos en el funcionamiento de la honeynet.

En el caso de tener una honeynet de primera generación, toda la recolección de datos recaería sobre el cortafuegos frontera de la red, por lo que los datos que podría recoger serían principalmente estadísticas de uso de la red y datos de los mecanismos de control de datos implementados en el cortafuegos. Aunque en algunos casos es posible la importación de datos externos a dichos cortafuegos, esta posibilidad es muy limitada y, además, la integración con herramientas de análisis de datos eficientes se ve muy limitada.

Para honeynets de segunda y tercera generación, la inclusión de un honeywall amplía la versatilidad de la honeynet para integrar mecanismos de colección de datos que admitan amplia variedad de datos y posibiliten la integración con herramientas de análisis de datos.

Actualmente, existen diferentes posibilidades para implementar los mecanismos del proceso de control de datos de la honeynet, entre otras:

- Graylog: software de administración de logs de código abierto gratuito para 5GB de datos al día. Si se desea más cantidad de datos y soporte técnico, se deben adquirir licencias de uso desde 60€ (precio que aumenta según las necesidades de despliegue).
- Graphite: software de recolección de datos de código abierto que se integra con muchas herramientas de captura de datos y herramientas de análisis de datos. La desventaja principal de Graphite es que el formato de ingesta de datos es único y las herramientas de captura de datos deben exportar en ese formato de datos para permitir la colección de estos.
- Logstash + Elasticsearch: software de colección de datos que forma parte del stack ELK (Elasticsearch + Logstash + Kibana) de código abierto desarrollado por la empresa Elastic.io. Elasticsearch almacena todos los datos en formato JSON para su posterior análisis por parte de herramientas de análisis, como Kibana. Para aquellos datos cuya exportación no se realiza de manera nativa en formato JSON, Logstash ofrece un punto intermedio de captura de datos en diferentes formatos para su traducción a formato JSON y envío a Elasticsearch.

8.4.4 Análisis de datos y administración

Tal y como se introduce en el apartado teórico, una honeynet debe proveer al administrador de la red de un proceso de análisis de datos eficiente que aporte valor a todos los datos generados por la honeynet, proceso sin el cual, los datos carecerían de valor.

Las herramientas de análisis de datos que mejor se adaptan al funcionamiento de una honeynet son aquellas que permiten al administrador de la red relacionar datos de diferentes

fuentes, buscar datos en diferentes espacios de tiempo, y todos aquellos procesos que faciliten la investigación de ataques informáticos.

Actualmente existen diferentes alternativas para el análisis de datos de diversas fuentes en una honeynet, entre las que se incluyen:

- Grafana: solución de análisis de datos de código abierto que se integra con diferentes fuentes de datos como, Elasticsearch, MySQL, Graphite, InfluxDB, etc. Es una buena alternativa si los datos a analizar contienen diferentes formatos. La desventaja principal de Grafana es que, al soportar gran cantidad de formatos de datos, el software de análisis de datos es pesado y requiere gran cantidad de procesamiento.
- Kibana: solución de análisis de datos de código abierto que conforma la tercera parte del stack ELK, junto con Elasticsearch y Kibana, desarrollada por la empresa Elastic.io. A la hora de ingesta datos procedentes de Elasticsearch, Kibana es la herramienta más optimizada y que ofrece las mejores herramientas de visualización y búsquedas de datos. La desventaja de Kibana es que solo ingesta datos de Elasticsearch y no de otras fuentes de datos diferentes.

Una vez que se han implementado todos los mecanismos de control, captura, colección y análisis de datos, la honeynet debe proveer al administrador de red de mecanismos de administración óptimos y seguros. Aunque la administración local siempre es una posibilidad, para facilitar el acceso remoto, la honeynet debería ser administrada a través del honeywall mediante mecanismos como túneles SSH, túneles VPN, paneles web cifrados mediante HTTPS, etc. También es recomendable separar todo este proceso de administración del caudal de tráfico que abre la honeynet a los atacantes, para que ningún atacante tenga visión del punto de entrada que el administrador utiliza para analizar los datos y gestionar la honeynet.

8.5 Software de los honeypots

8.5.1 Nivel de interacción

En el apartado teórico se establecen tres diferentes niveles de interacción para los honeypots, según las acciones que el atacante podía realizar contra el honeypot en cuestión: baja, media y alta.

El despliegue de servicios de baja interacción supone menos dificultad de configuración, aunque aumentan la posibilidad de que el atacante detecte la existencia del honeypot, debido a la poca posibilidad de interacción que el honeypot ofrece. Los servicios de media interacción aumentan un poco la complejidad de configuración, aunque también disminuye la posibilidad de detección, ya que las acciones que un atacante puede realizar contra el servicio son menos limitadas. Sin embargo, los servicios de alta interacción son los óptimos a la hora de esconder el verdadero propósito de estos, ya que, al fin y al cabo, son servicios reales mal configurados o con vulnerabilidades conocidas por el administrador de la red. También cabe decir que la configuración de servicios vulnerables de alta interacción es más complicada, debido a que un servicio real vulnerable no limita las acciones que un atacante puede realizar contra un honeypot, así que, dentro la inseguridad que ofrece un servicio

vulnerable, el honeypot debe ser configurado de tal manera que limite las acciones más destructivas que podría realizar un atacante contra el sistema.

La elección del nivel de interacción de los servicios que ofrece un honeypot y, en este caso, la honeynet, dependerá de las preferencias y necesidades de cada honeynet.

8.5.2 Servicios vulnerables

Cualquier servicio real que se encuentre en cualquier entorno de producción de una empresa puede ser replicado en un honeypot para investigar los diferentes vectores de ataque que puede sufrir el servicio y aplicar dicho conocimiento a la protección del servicio real. La elección del servicio adecuado depende del contexto de despliegue del proyecto. En el caso de un proyecto desplegado en un entorno empresarial, interesa la replicación de servicios reales, mientras que, en el caso de un entorno de investigación, el servicio elegido dependerá de las necesidades de investigación en el momento concreto: análisis de nuevas tecnologías, análisis de capacidades de respuesta de sistemas antes situaciones de ataque, etc.

Una vez que se eligen los servicios que se desean replicar y/u ofrecer en una honeynet, existen multitud de diferentes implementaciones de servicios vulnerables en todos los niveles de interacción. Se deberán seleccionar los servicios que faciliten la tarea de captura de datos y posterior colección y análisis, así como dificulten la tarea de identificación de la honeynet por parte de los atacantes.

9 Descripción de la solución propuesta

Nota: Los detalles específicos de las configuraciones de las herramientas especificadas en este apartado se encuentran en el ANEXO A: CONFIGURACIONES.

9.1 Arquitectura de la honeynet

Tal y como se puede observar en los antecedentes del presente proyecto, la red interna de la Diputación de Cádiz, administrada por EPICSA, es una red extensa, compleja y muy heterogénea, desde el núcleo de la red, hasta la zona frontera de la misma con toda la interconexión a internet, sedes remotas y teletrabajadores de la Diputación de Cádiz.

Un despliegue de una honeynet de tercera generación queda descartado para el presente proyecto debido a que la red es tan compleja, que la investigación que supone la integración de servicios vulnerables, así como protección de los sistemas colindantes, sería de tal magnitud que se aleja del alcance del proyecto en cuestión. Por ello, se opta por implantar una honeynet separada física y lógicamente de la red interna en producción de la Diputación de Cádiz.

Una honeynet de primera generación supondría configurar todo el control, captura, colección y análisis de datos en los cortafuegos frontera ya desplegados en la red de la Diputación de Cádiz. Esta tarea puede suponer un gran reto a la hora de que dicha configuración no suponga ningún tipo de interferencia con el desempeño diario de la red interna de la Diputación de Cádiz, ya que, sin la honeynet, los cortafuegos de la frontera de la red ya soportan una gran carga de trabajo con los accesos a la red DMZ y con el control de accesos a las redes internas.

Por ello, para cumplir con el requisito **R-01: Desplegar una honeynet que no sobrecargue la infraestructura existente** de este proyecto, se va a desplegar una honeynet de segunda generación, donde todas las tareas de control, captura, colección y análisis de datos recaigan en el honeywall, liberando así a los cortafuegos de la frontera de toda la posible carga de trabajo que supondría la implementación de toda la gestión de datos de la honeynet. El diagrama de la red frontera de la Diputación de Cádiz, tras la integración de la honeynet sería el siguiente:

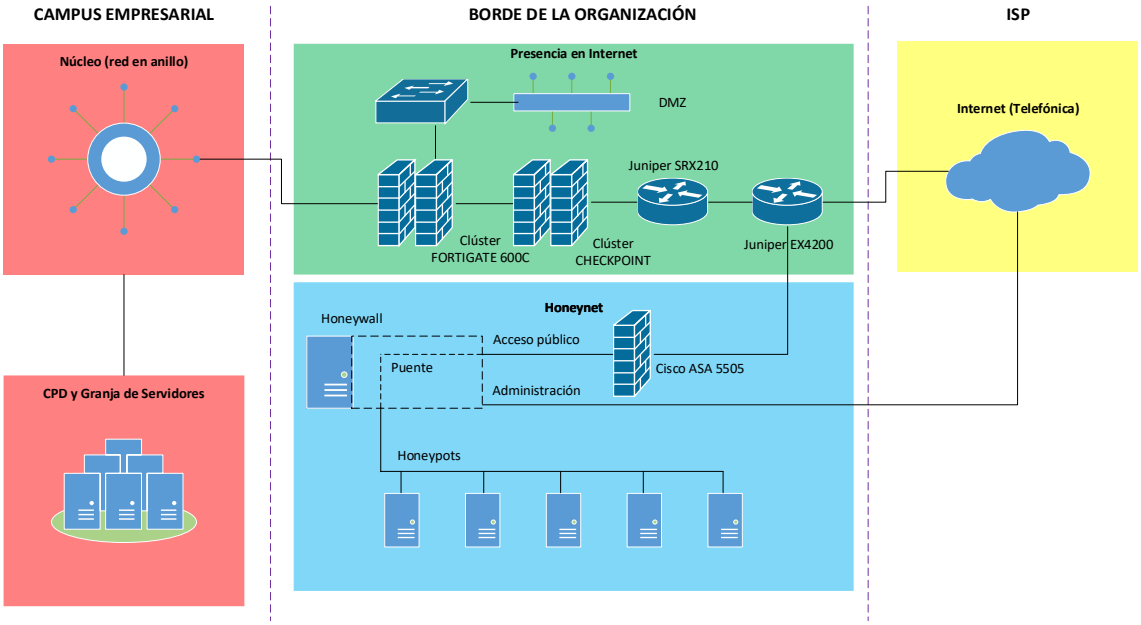

Figura 7 Solución honeynet de segunda generación

El cortafuegos CISCO ASA 5505 simplemente deberá permitir la entrada y salida de la honeynet hacia Internet y viceversa. La red interna y la honeynet no tendrán visibilidad alguna entre ellas.

9.2 Despliegue de la honeynet

Para reducir al máximo posible los costes de despliegue de la honeynet mientras que se aprovecha al máximo las capacidades de los softwares actuales de virtualización disponibles, la honeynet se va a desplegar en la red de la Diputación de Cádiz mediante soluciones virtualizadas.

En concreto, una honeynet virtualizada independiente supondría la adquisición de hardware de altas prestaciones al tener que implementar todos los procesos de la honeynet en la misma máquina y, aunque la haría más portable, no es conveniente que la honeynet depende de un único punto de fallo en el hardware sobre el que se fuera a desplegar. Además, los ataques que pudieran sufrir los servicios vulnerables en máquinas virtuales podrían tener interferencias con el desempeño del honeywall, impidiendo el control, la captura, la colección y el análisis de datos.

Por ello, para cumplir con el requisito **R-02: Desplegar una honeynet escalable.** de este proyecto, se va a desplegar una honeynet virtualizada híbrida donde se configuran dos servidores, uno para el honeywall, y otro para los honeypots virtualizados. Así subsanamos algunas desventajas introducidas anteriormente. Por una parte, estamos reduciendo los

requerimientos hardware de los servidores al repartir la carga de trabajo entre ellos. También eliminamos el único punto de fallo de la honeynet, permitiendo que un ataque en uno de los honeypots no tenga influencia directa en el honeywall, permitiendo así todos los procesos de control, captura, colección y análisis de datos. De igual modo, al desplegar los honeypots de manera virtualizada, se aumenta al máximo la rapidez en el despliegue de nuevos servicios vulnerables.

Con tal consideración, el diagrama de red de la red frontera de la Diputación de Cádiz con una honeynet de segunda generación introducido en el apartado anterior se ve actualizado para introducir una honeynet virtual híbrida.

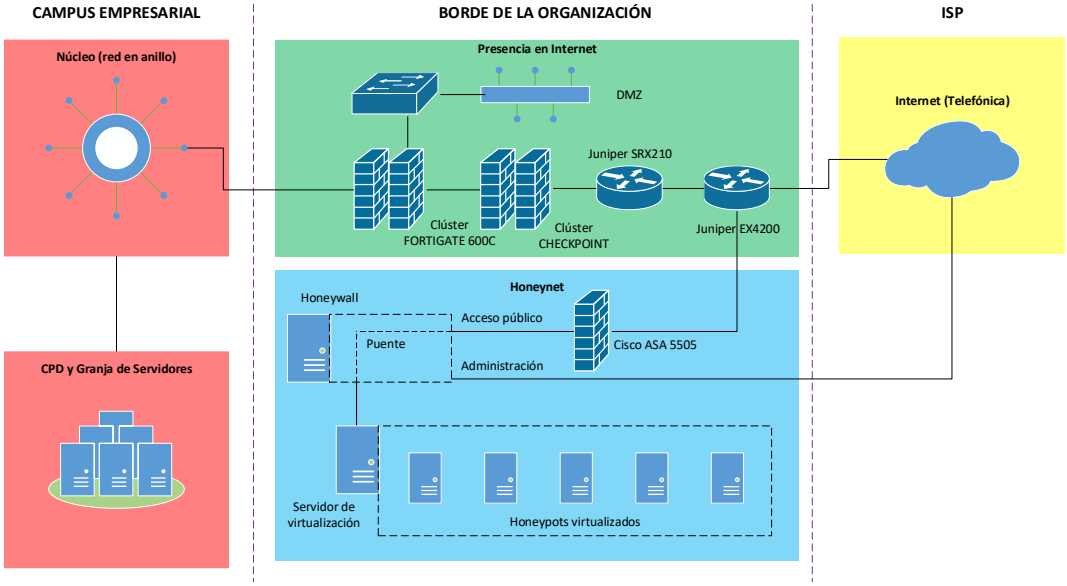

Figura 8 Solución de honeynet virtual híbrida

El acceso público a la honeynet se establece a través del caudal contratado con Telefónica para datos y aplicaciones, así como para el acceso público a la DMZ de la red de la Diputación de Cádiz. El acceso de administración a la honeynet se realiza a través de una línea exclusiva ADSL de 12 Mb/s contratada con el ISP Telefónica.

Las direcciones asignadas a cada uno de los componentes de la honeynet son las siguientes:

- Acceso público: 194.179.87.22. Con lo cual, el acceso público a la honeynet se integra con los demás servicios ofrecidos por EPICSA en las diferentes direcciones de los bloques de direcciones públicas que se le han asignado.
- Acceso para administración: 83.37.127.151 mediante la línea exclusiva contratada con Telefónica.
- Direccionamiento interno: 10.0.0.0/24. Todos los equipos internos de la honeynet, tanto el servidor de virtualización, el propio honeywall y los honeypots tendrán direcciones IP internas pertenecientes al bloque de direcciones establecido.

9.2.1 Software de virtualización

Tal y como se ha establecido en el apartado anterior, la arquitectura de la honeynet de segunda generación a desplegar en la frontera de la red de la Diputación de Cádiz se implementará como una solución virtual híbrida, donde se virtualizarán los honeypots.

Para la implementación de la honeynet virtual híbrida se debe tener en cuenta la apuesta continua que la Diputación de Cádiz tiene por el uso de software de código abierto en sus proyectos desarrollados internamente, por lo que, una solución privativa como VMWare no es aplicable. Para elegir entre VirtualBox y Proxmox VE debemos tener en cuenta de que, todos los honeypots virtualizados en la honeynet deben ser fácilmente administrables de manera remota. VirtualBox no ofrece las capacidades necesarias para administración remota, quedándose en el uso del software a través de la consola de comandos. Por el contrario, Proxmox VE nos ofrece todas las herramientas necesarias para configurar, desplegar y administrar máquinas virtuales de honeypots de la forma más eficiente posible, añadiendo funcionalidades extra como:

- Administración web, cifrada mediante el protocolo HTTPS.
- Exportación de logs, tanto del servidor, como de las distintas máquinas virtuales.

Figura 9 Logo de Proxmox

Para el acceso de administración al servidor de virtualización Proxmox, se le ha asignado la dirección IP 10.0.0.3/24, perteneciente al bloque de direcciones internas de la honeynet.

9.3 Hardware de despliegue

Teniendo en cuenta que se ha establecido el despliegue de una honeynet virtual híbrida de segunda generación en la zona frontera de la red de la Diputación de Cádiz, se han elegido dos servidores iguales: uno para el honeywall y otro para el servidor de virtualización Proxmox. Ambos servidores poseen los requisitos suficientes para soportar todos los procesos de control, captura, colección y análisis de datos en el honeywall y todo el proceso de creación y gestión de máquinas virtuales en el servidor de virtualización.

El modelo de servidor elegido es el HPE ProLiant DL360 Gen9 Intel Xeon E5-2620v4. A continuación se detallan sus características técnicas.

Figura 10 Servidor HPE ProLiant DL360 Gen9 Intel Xeon E5-2620v4

Concepto	Valor
Procesador	Intel Xeon E5 v4 E5-2620v4 8 núcleos 2,1 GHz

Memoria RAM	16 GB DDR4-SDRAM
Capacidad máxima HDD	20 TB
Conexiones	4x Gigabit Ethernet 1x VGA 1x Serie 4x USB 3.0
Alimentación	500W
Chasis	1U

Tabla 2 Especificaciones HPE ProLiant DL360 Gen9 Intel Xeon E5-2620v4

9.4 Software del honeywall

9.4.1 Control de datos

Tal y como se ha especificado con anterioridad, se va a desplegar una honeynet de segunda generación virtual híbrida, lo que conlleva que todo el control de datos y todos los mecanismos que se necesitan para su desempeño se deben implementar en el honeywall. Para aumentar al máximo la compatibilidad del honeywall y del software de control de datos con otros mecanismos de datos del honeywall (captura, colección y análisis) se opta por la opción de configurar en el honeywall un sistema operativo GNU/Linux con la utilidad IPTables.

Se va a configurar específicamente un sistema operativo Ubuntu Server 16.04.3 de 64 bits. Como la utilidad de IPTables está instalada por defecto en este sistema operativo, no hace falta ninguna instalación extra de momento.

Al puente de red del honeywall se le asigna la dirección IP interna 10.0.0.2/24 del bloque de direcciones internas del honeywall.

9.4.1.1 Limitación de conexiones

Para el cumplimiento del requisito **R-03: Implementar control y limitación de conexiones**, se configura IPTables para que limite las conexiones entrantes en el puente de red a 50 paquetes por segundo, de tal manera, serán descartadas aquellas conexiones que superen dicho número de paquetes por segundo. Cada segundo se reinicia el contador de paquetes por segundo, por lo que las conexiones no se bloquean permanentemente, sino que se limitan, por lo que un atacante no pierde la conexión con la honeynet, sino que se limita su capacidad de llevar a cabo acciones que supongan una alta carga de tráfico.

El esquema lógico de configuración de IPTables con limitación de tráfico sobre el puente de red sería el siguiente.

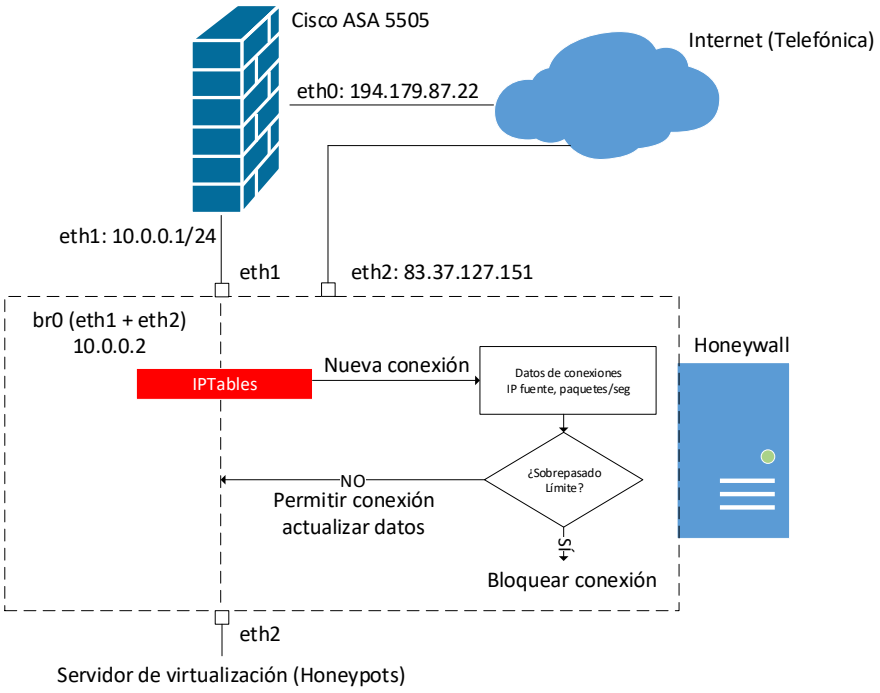

Figura 11 Funcionamiento de IPTables sobre el puente de red

La aplicación del control de conexiones se aplica tanto para conexiones iniciadas desde el exterior como para conexiones iniciadas por alguno de los honeypots o incluso el servidor de virtualización.

Configuración en ANEXO A: página 100.

9.4.1.2 Prevención de intrusiones en red

Focalizando el análisis de soluciones en la instalación de un IPS en la honeynet, para el cumplimiento del requisito **R-04: Implementar prevención de intrusiones en red**, al encontrarnos en una honeynet de segunda generación se debe tener en cuenta que el IPS se deberá desplegar en el honeywall por lo que ahora se debe decidir que software escoger entre todas las alternativas introducidas con anterioridad. Debido a la continua apuesta de la Diputación de Cádiz por el código abierto, se descarta cualquier opción privativa. Dicho esto, entre Snort y Suricata, las dos soluciones IPS de código abierto más conocidas, se opta por Suricata por las siguientes razones:

- Soporte de ejecución multihilo, lo que conlleva mejor rendimiento.
- Exportación de datos en formato JSON, lo que facilita la integración de dichos datos con los mecanismos de colección de datos de la honeynet.

A la hora de configurar Suricata para su desempeño como IPS en una honeynet sobre un puente red, se deben tener en cuenta las siguientes consideraciones.

- En su modo IPS, Suricata utiliza un sistema de colas de paquetes denominado NFQ. Un sistema Linux puede decidir enviar los paquetes de red mediante IPTables a diferentes colas NFQ para que sean procesados por otro software diferente del usuario. Suricata es capaz de escuchar en diferentes colas NFQ y decidir si el tráfico puede continuar su camino o ser cortado.

- Se debe habilitar la actualización automática de reglas para que esta recojan la información del tráfico de red malicioso más actualizada posible, de tal manera que el tráfico que atraviese la red sea, o no malicioso, o de nuevas amenazas sin descubrir. Para este proyecto, se descargarán las reglas de EmergingThreats [8] mediante el software Oinkmaster [9].
- Las reglas de EmergingThreats, por defecto, solo alertan del tráfico no malicioso en vez de bloquearlo. Para cambiar este comportamiento al deseado, se deben modificar las reglas cada vez que se actualicen, para ello, se utilizará también el software Oinkmaster.

La modificación de reglas para cambiar su comportamiento consiste en sustituir el campo de acción que determina que debe hacer Suricata si el tráfico analizado coincide con lo establecido en la regla. Si el campo está establecido en *alert*, Suricata alerta y deja pasar el tráfico, por el contrario, si el campo está establecido en *drop*, Suricata alerta y bloquea el tráfico, por lo tanto, todas las reglas que contengan el campo *alert* y se desea que bloqueen el tráfico, se debe cambiar por *drop*. Pero no se pueden cambiar todas las reglas, debidos a que existe una clasificación de las reglas en tres niveles de peligro: alto, medio y bajo.

Bajo	Medio	Alto
icmp-event	web-application-activity	attempted-admin
misc-activity	unusual-client-port-connection	attempted-user
network-scan	system-call-detect	inappropriate-content
not-suspicious	suspicious-login	policy-violation
protocol-command-decode	suspicious-filename-detect	shellcode-detect
string-detect	successful-recon-limited	successful-admin
unknown	successful-recon-largescale	successful-user
tcp-connection	successful-dos	trojan-activity
	rpc-portmap-decode	unsuccessful-user
	non-standard-protocol	web-application-attack
	misc-attack	
	default-login-attempt	
	denial-of-service	
	bad-unknown	
	attempted-recon	
	attempted-dos	

Tabla 3 Categorías de reglas de Suricata

Si se bloquea el tráfico que es identificado por reglas de categoría baja, se corre el riesgo de bloquear tráfico legítimo. Por lo tanto, se decide modificar las reglas de nivel medio y alto. Para ello, se configurará Oinkmaster para que cambie la acción de todas aquellas reglas de nivel medio y alto, salvo las de violación de políticas (policy-violation, tipo Alto), que deben ser especificadas por la organización en cuestión.

Una vez desplegado Suricata en el honeywall, el diagrama de control de datos del honeywall es el siguiente

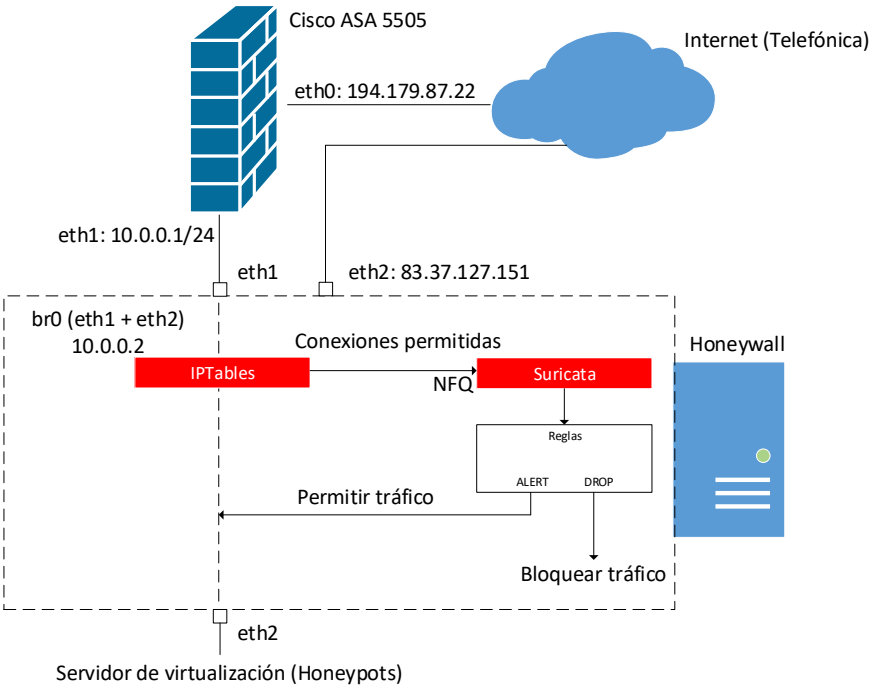

Figura 12 Suricata en el control de datos del honeywall

La aplicación de reglas de Suricata en modo IPS al tráfico de red se realiza para el tráfico del puente de red en ambos sentidos, Internet hacia honeypots y viceversa. En cada alerta que Suricata genere, ya sea para dejar pasar el tráfico o para bloquearlo, aparecerán los siguientes datos principales:

- Regla que genera la alerta
- Categoría de la regla
- Acción tomada sobre el tráfico
- Direcciones IPs y puertos relacionados
- Payload del paquete

Un ejemplo de alerta generada por Suricata con la ya mencionada configuración sería la siguiente:


```
{
   "timestamp":"2018-06-04T00:23:27.008914+0200",
   "flow_id":537750460,
   "event_type":"alert",
   "src_ip":"61.177.172.47",
   "src_port":12564,
   "dest_ip":"10.0.0.4",
   "dest_port":22,
   "proto":"TCP",
   "alert":{
      "action":"allowed",
      "gid":1,
      "signature_id":2019876,
      "rev":4,
      "signature":"ET SCAN SSH BruteForce Tool with fake PUTTY version",
      "category":"Detection of a Network Scan",
      "severity":3
   },
   "ssh":{
```

```
      "client":{
         "proto_version":"2.0",
         "software_version":"PUTTY"
   },
   "payload":"U1NILTIuMC1QVVRUWQ0K",
   "payload_printable":"SSH-2.0-PUTTY\r\n",
   "stream":1,

"packet":"RQAAQ2efQAAuBvExPbGsLwoAAAQxFAAW+jLy8uzKc7SAGADlKsYAAAEBCAoAY2KkDDJje1NTSC0
yLjAtUFVUVFkNCg=="
}
```


En dicha alerta se puede observar toda la información que se ha especificado anteriormente, en concreto, es una alerta sobre un ataque de fuerza bruta al protocolo SSH de la IP interna 10.0.0.4 desde la IP externa 61.177.172.47, simulando un inicio de sesión desde la herramienta *Putty*.

Además, a parte de las alertas generadas por la coincidencia de tráfico con las reglas definidas en los archivos de reglas, se configura Suricata para que alerte y permita el tráfico cuando detecte eventos genéricos en la red, tales como:

- Sesiones HTTP/HTTPS.
- Sesiones SSH.
- Búsquedas DNS.
- Transferencia de archivos.
- Envío de correos mediante SMTP.

Un ejemplo de una alerta generada por estos eventos genéricos ante una búsqueda DNS de un honeypot sería:


```
{
   "timestamp":"2018-06-06T05:11:38.882335+0200",
   "flow_id":2742435456,
   "event_type":"dns",
   "src_ip":"8.8.8.8",
   "src_port":53,
   "dest_ip":"10.0.0.4",
   "dest_port":36390,
   "proto":"UDP",
   "dns":{
      "type":"answer",
      "id":43329,
      "rcode":"NOERROR",
      "rrname":"api.snapcraft.io",
      "rrtype":"A",
      "ttl":184,
      "rdata":"91.189.92.19"
   }
}
```


En dicha alerta se puede observar la respuesta de tipo A del servidor DNS 8.8.8.8 (Google) a la IP interna 10.0.0.4 sobre la dirección IP del dominio "api.snapcraft.io", que sería 91.189.92.19

Estos eventos, en una red en producción normal no tendrían por qué suponer un ataque informático, pero en una honeynet, todo tráfico es sospechoso, por lo que toda información que se pueda conseguir siempre es útil.

Los logs propios que genere Suricata relacionados con sus estadísticas de procesamiento y eventos internos se almacenarán en el honeywall en el directorio */var/log/suricata*.

Configuración en ANEXO A: página 101.

9.4.2 Captura de datos

Para implementar todos los mecanismos de captura de datos en la honeynet para capturar todos los datos descritos en el apartado de alternativas, se va a necesitar de la configuración de múltiples solucione software, que se detallan a continuación.

9.4.2.1 Detección de intrusiones en honeypots

Para el cumplimiento de los requisitos **R-05: Implementar detección de intrusiones en honeypots**, **R-06: Implementar control de integridad en honeypots** y **R-07: Implementar registro de inicios de sesión**, se configurará la infraestructura de HIDS del software OSSEC en modo cliente/servidor, siendo el servidor el honeywall y los clientes los diferentes honeypots que se instalen en la honeynet. Las clientes OSSEC de los honeypots enviarán al servidor los datos cifrados de los logs monitorizados y el servidor, juzgará los eventos según un conjunto de reglas para decidir si alertar o no ante dicho evento recibido. El funcionamiento esquemático es el siguiente:

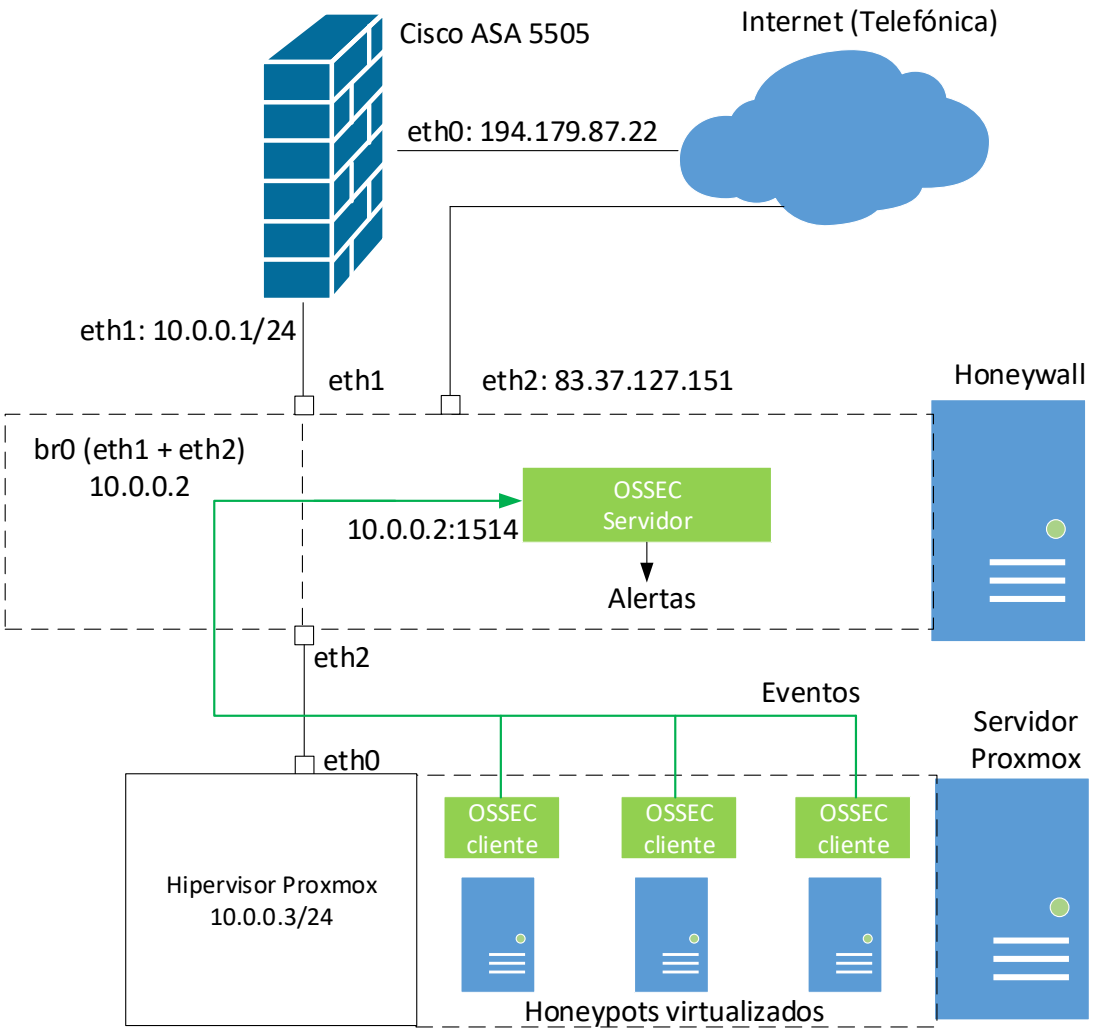

Figura 13 Arquitectura de HIDS OSSEC cliente/servidor

Los datos que se almacenan ante una alerta del servidor generada por el envío de un evento sospechoso de un honeypot son principalmente:

- Regla que genera la alerta.
- IP origen de la actividad sospechosa (si procede).
- Usuario involucrado (si procede).
- Origen del evento que genera la alerta (honeypot + fichero local).
- Log completo del evento.

Un ejemplo de alerta generada por un evento en un honeypot puede ser el inicio de sesión por SSH del usuario *root*:


```
{
   "rule":{
      "level":3,
      "comment":"SSHD authentication success.",
      "sidid":5715
   },
   "srcip":"150.9.23.7",
   "dstuser":"root",
   "location":"(hp1) 10.0.0.4->/var/log/auth.log",
   "full_log":"Jun  6 01:02:01 test-ubuntu sshd[25274]: Accepted publickey for root
from 150.9.23.78 port 41932 ssh2: RSA
SHA256:dsdbKyMfIq40BkXigBRGKTEkbLr5511n3z39q5zpz2A"
}
```


En la información de la alerta se puede ver como el honeypot HP1 con dirección IP 10.0.0.4 ha enviado un evento del fichero *auth.log* y el servidor ha lanzado la alerta de autenticación correcta por SSH para el usuario *root* en dicho honeypot.

Una alerta generada por OSSEC ante un cambio de fichero en un sistema de archivos puede ser la siguiente:


```
{
   "rule":{
      "level":3,
      "comment":"Log file rotated.",
      "sidid":591
   },
   "location":"(hp1) 10.0.0.4->ossec-logcollector",
   "full_log":"ossec: File rotated (inode changed): '/var/log/syslog'."
}
```


En este caso, OSSEC está alertando de que el fichero syslog del honeypot HP1 ha cambiado debido a una rotación de logs programada en dicho honeypot.

Configuración en ANEXO A: página 105.

9.4.2.2 Registro de comandos en honeypots

Para el cumplimiento del requisito **R-08: Implementar registro de comandos**, en los sistemas GNU/Linux existe una utilidad llamada *script*, que graba en un fichero de texto las sesiones que se establecen en un servidor, ya sea por consola, por Telnet, por SSH, etc. Dicha utilidad se configurará en todos los honeypots basados en GNU/Linux que se desplieguen en la honeynet.

Dicha utilidad se lanza cada vez que un usuario entra en el sistema o cada vez que se inicia sesión en una cuenta diferente a la del usuario inicial en el honeypot. La utilidad generará los ficheros de sesión con el formato:

```
script.{fecha}.{honeypot}.{usuario}.log
```

Siendo:

- Fecha: fecha de inicio de sesión, en formato:

  Año(yyyy)-Mes(MM)-Día(dd)-Hora(hh)-Minuto(mm)-Segundos(ss)

- Honeypot: nombre del honeypot.
- Usuario: nombre del usuario con el que se inicia sesión

Por ejemplo, si en un honeypot llamado *test-ubuntu* el usuario *test* inicia sesión por SSH, el fichero generado sería el siguiente:

/var/log/script/script.2018-06-06-09-39-33.test-ubuntu.test.log

Dentro se podría ver que ha hecho el usuario dentro del honeypot, en este caso, un simple listado de ficheros:

```
# cat /var/log/script/script.2018-06-06-09-39-33.test-ubuntu.test.log
       Script iniciado (mié 06 jun 2018 11:39:33 CEST)
       bash-4.3$ ls
       files
       bash-4.3$ exit
       exit
```

9.4.2.3 Captura de tráfico de la honeynet

Para el cumplimiento del requisito **R-09: Implementar captura de tráfico**, se configura el software *tcpdump* con las siguientes características:

- Captura continua del tráfico que atraviesa el puente de red.
- Rotación de los archivos de captura de tráfico:
  - Cada hora o cuando el archivo supera los 2GB de tamaño.
  - Se mantienen los archivos del último mes.

La herramienta *tcpdump* recoge el tráfico permitido por IPTables y por Suricata, de tal manera que queda libre de todo el tráfico no deseado en la honeynet.

Configuración en ANEXO A: página 106.

9.4.2.4 Estadísticas de uso y sesiones de la red

Para el cumplimiento del requisito **R-10: Recoger estadísticas de uso y sesiones de la red**, se configurará en el honeywall la utilidad *fprobe* de generación de estadísticas de red para la recogida de estadísticas de uso de la honeywall a través del tráfico que atraviesa el puente de red, así como de las sesiones establecidas. La herramienta almacenará los datos relacionados con las sesiones que se establezcan en la honeynet:

- IPs y puertos origen/destino de la sesión.
- Hora de inicio de la sesión.

- Bytes de datos transferidos en la sesión.

La herramienta *fprobe* recoge el tráfico permitido por IPTables y por Suricata, de tal manera que queda libre de todo el tráfico no deseado en la honeynet.

Configuración en ANEXO A: página 106.

9.4.2.5 Estadísticas de recursos de la honeynet

Para el cumplimiento del requisito **R-11: Recoger estadísticas de uso de recursos de la honeynet**, primero, se configura en el honeywall la utilidad *collectd* de recogida de estadísticas de uso en sistemas GNU/Linux. Dicho software recogerá estadísticas sobre

- Uso de CPU
- Uso de memoria RAM
- Operaciones de lectura/escritura de disco.

Para los diferentes honeypots virtualizados en la honeynet, el servidor de virtualización Proxmox ofrece la posibilidad de exportar las estadísticas de uso tanto del servidor, como de los honeypots virtualizados, entre las que se incluyen:

- Uso de CPU
- Uso de memoria RAM
- Operaciones de lectura/escritura de disco.

Configuración en ANEXO A: página 106

9.4.3 Colección de datos

9.4.3.1 Cifrado de datos

El requisito establecido en este proyecto para la colección de datos es el requisito **R-12: Implementar cifrado en la colección de datos**. Las diferentes comunicaciones para el traspaso de archivos de logs entre los honeypots y el honeywall debe ser cifrada. El traspaso de logs que realiza OSSEC en su infraestructura de cliente/servidor se realiza de manera cifrada mediante un sistema de clave única compartida.

Se implementa otro mecanismo de colección de datos para la salvaguarda de los logs de los honeypots basados GNU/Linux, que es, mediante SSH y desde el honeywall, copiar todo el sistema de ficheros que cuelga del directorio */var/log* de los honeypots, donde residen todos los logs generados por el software instalado en estos. Dicha copia se realiza de manera unidireccional con la herramienta *Rsync*. *Rsync* utiliza conexiones SSH para mantener una sincronización de los logs que se deseen. Dicha copia se realiza mediante un par de claves pública/privada generada en el honeywall cada minuto, mediante una tarea *cron*. La clave pública del honeywall está autorizada en los honeypots, de manera que la conexión en dicho sentido es permitida, pero las claves de los honeypots no se permiten en el honeywall, por lo que el sentido inverso de la conexión no se permite. Los logs de los diferentes honeypots se guardarán en el honeywall en */var/log/hp{1, 2,...,n}*, siendo n el identificador específico de cada honeypot.

Configuración en ANEXO A: página 107.

9.4.3.2 Centralización del almacenamiento de logs

Tal y como se especifica en el apartado de análisis y alternativas de soluciones de este proyecto, se debe implementar en el honeywall una herramienta de colección de datos que centralice el almacenamiento de datos para su posterior análisis por diferentes herramientas.

Para este proyecto, se opta por la utilización del stack ELK (elasticsearch, logstash y kibana) por los siguientes motivos:

- Todo el stack de software es de cógido abierto.
- Logstash admite la ingesta de datos de múltiples formatos y fuentes diferenciadas.
- Todo el stack de software esta optimizado para su funcionamiento conjunto.

El funcionamiento de logstash con elasticsearch es el siguiente, de manera esquemática:

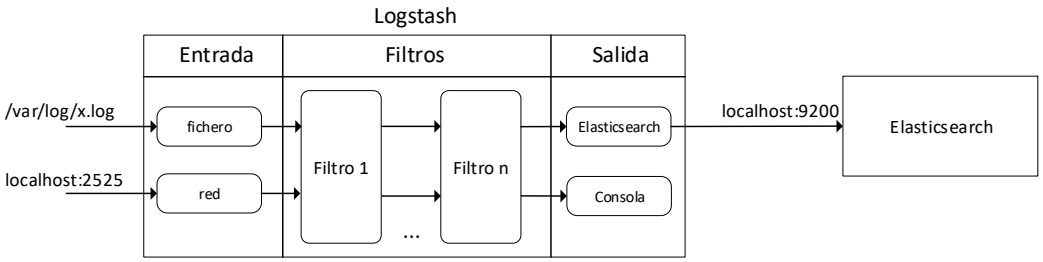

Figura 14 Funcionamiento de logstash con elasticsearch

El software Logstash es capaz de recoger datos de numerosas y diversas fuentes mediante plugins, entre las que se incluyen, leer de un fichero de logs estándar y escuchar en un puerto de red para capturar los datos recibidos. Independientemente de dónde se reciban los datos, por un fichero, por red, etc., Logstash, es capaz, mediante codecs, interpretar los datos recibidos para su restructuración en formato JSON, en el caso de que los datos no se reciban en dicho formato. Entre estos codecs, se encuentran, por ejemplo:

- Collectd: interpretar datos del software collectd.
- Netflow: interpretar datos de estadísticas de red.
- Plain: interpretar datos en texto plano.

Una vez interpretados los datos y restructurados en formato JSON, Logstash aplica una seria de filtros definidos por el usuario, entre los que se pueden encontrar:

- Borrado/adición de campos de datos.
- Modificación de datos.
- Cifrado de datos.
- Creación de marcas temporales.
- Búsquedas DNS.

Cuando se han aplicado todos los filtros y los datos ya se han terminado de restructurar y adaptar a formato JSON, Logstash es capaz de enviar los datos a numerosos destinos, entre los que se encuentran:

- Elasticsearch.
- Consola.

- Correo electrónico.
- Fichero.

Si se desea almacenar los datos para su posterior análisis, lo usual es almacenar dichos datos en Elasticsearch. Elasticsearch es un motor de búsqueda de datos basado en Apache Lucene, una API de código abierto de recuperación de información. Elasticsearch ofrece una API RESTful para la búsqueda de datos almacenados, de tal manera que se facilita su integración con diversos lenguajes de programación y /o herramientas de análisis, tales como Kibana o Grafana.

En el honeywall, Logstash recibirá los datos de las herramientas de control y captura de datos especificadas en apartados anteriores por los siguientes medios:

- Suricata, a través del fichero de alertas en JSON en */var/log/suricata/eve.json*
- OSSEC, a través del fichero de alertas en JSON en */var/ossec/logs/alerts/alerts.json*
- Fprobe, a través del puerto de red 2055.
- Collectd, a través del puerto de red 25826.
- Proxmox, a través del puerto de red 2003.

Para que Elasticsearch pueda identificar las diferentes fuentes de datos que se almacenan en la base de datos, se le debe indicar un índice a cada fuente. Para ello, Logstash le adjunta dicho índice a cada dato. Para este proyecto, los índices serán los siguientes:

- Suricata: logstash-suricata-{fecha}
- OSSEC: logstash-ossec-{fecha}
- Fprobe: logstash-netflow-{fecha}.
- Collectd: logstash-collectd-{fecha}.
- Proxmox: logstash-proxmox-{fecha}.

El funcionamiento esquemático de Logstash y Elasticsearch en conjunto con las herramientas de control y captura de datos es el siguiente:

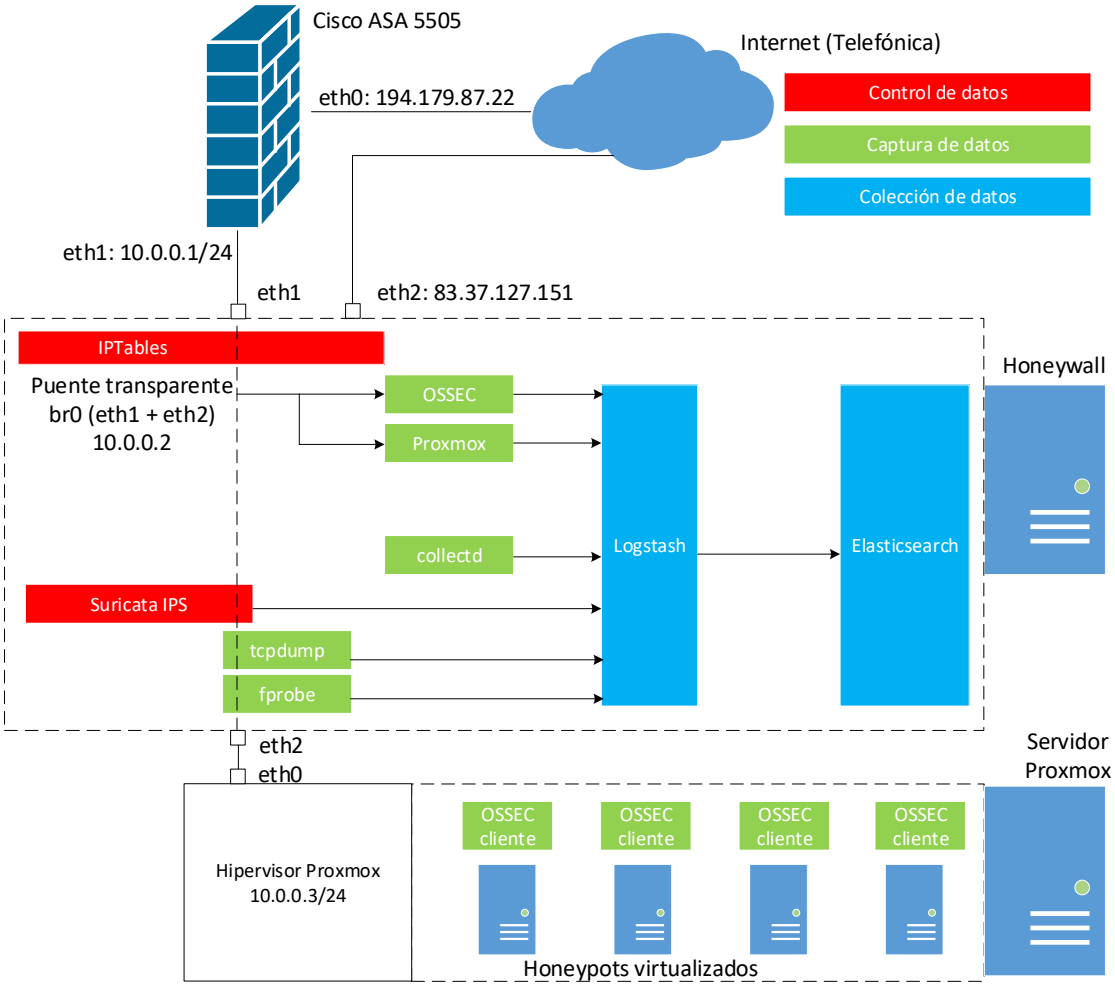

Figura 15 Integración de control, captura y colección de datos en el honeywall

Se puede observar como las herramientas de control y captura de datos envían sus datos a Logstash mediante los medios ya mencionados, y como Logstash envía los datos a Elasticsearch para su almacenamiento y posterior análisis.

Configuración en ANEXO A: páginas 107 y 108.

9.5 Análisis de datos y administración de la honeynet

9.5.1 Administración por canales cifrados

Para el cumplimiento del requisito **R-14: Configurar una conexión cifrada con el honeywall** de este proyecto, se deben configurar todos los mecanismos necesarios para que las comunicaciones que se lleven a cabo para administrar el honeywall y/o analizar los datos se realicen de manera cifrada y segura.

Para ello, primero, se separa todo el tráfico de administración de la honeynet con el tráfico malicioso de la honeynet mediante una interfaz de red separada. Dicha interfaz de red tendrá un direccionamiento IP público asignado para su acceso remoto y un caudal de tráfico separado con una línea ADSL de telefónica de 12 Mb/s.

Se configura una red privada virtual (VPN) de punto a punto con la interfaz de mantenimiento del honeywall mediante un sistema de claves estáticas que crea una red

privada con direccionamiento IP 10.8.0.1/24 con el software OpenVPN, siendo el honeywall el equipo con la dirección 10.8.0.1/24 asignada. Los logs de conexión a la VPN se registran en el honeywall en */etc/openvpn/openvpn-status.log*.

Configuración en ANEXO A: página 111.

Además, se configura un servidor SSH en el honeywall para su acceso administrativo, siendo la interfaz VPN, el único punto de entrada permitido por dicho servidor:

Acceso SSH a honeywall: 10.8.0.1:22

Los logs de acceso al servidor SSH para administración como de las conexiones para sincronización de logs entre honeypots y honeywall se registran en el honeywall en */var/log/auth.log*.

Configuración en ANEXO A: página 111.

Para el acceso administrativo al servidor de virtualización Proxmox, se configura en el honeywall una redirección mediante reglas IPTables para el acceso a través de la interfaz VPN del honeywall:

Acceso Proxmox: https/10.8.0.1:9999 -> redirigido a Proxmox -> https://10.0.0.3/

Configuración en ANEXO A: página 100 (sección nat).

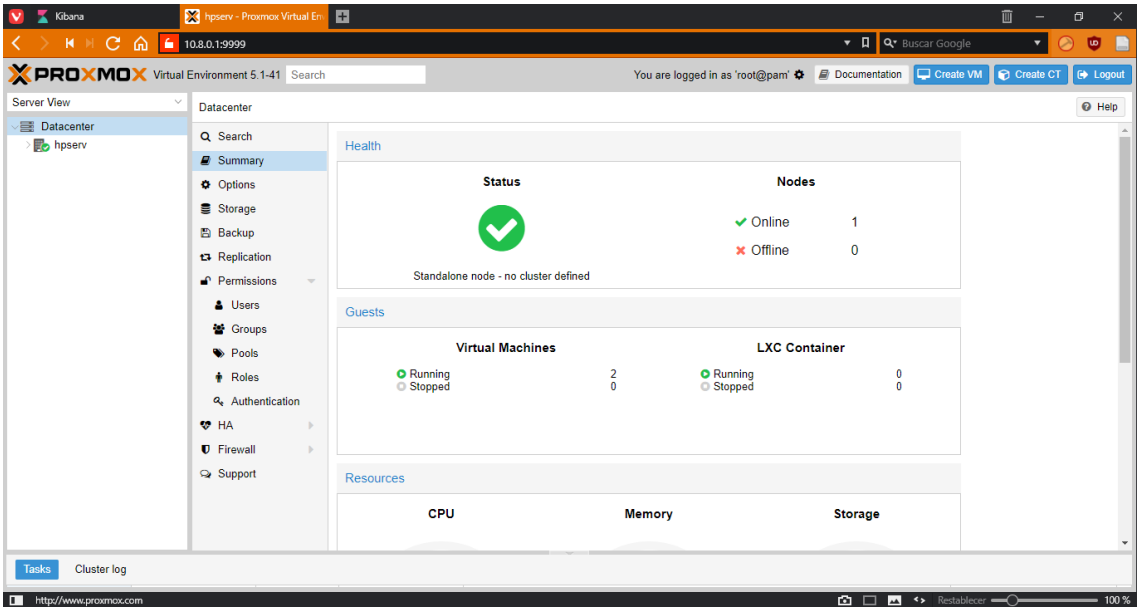

Figura 16 Acceso a la interfaz web del servidor Proxmox

9.5.2 Análisis de datos

Para el cumplimiento del requisito **R-13: Implementar un sistema de análisis de información** de este proyecto se debe configurar en el honeywall las herramientas necesarias para el análisis de todos los datos almacenados en Elasticsearch gracias a todos los mecanismos de control, captura y colección de datos ya especificados en apartados anteriores.

Para este proyecto, se va a optar por la configuración de la herramienta Kibana, en detrimento de la herramienta Grafana, debido a que Kibana, al ser Elasticsearch el único motor de búsqueda de datos disponible, es la herramienta que ofrece mejores resultados y búsquedas más avanzadas.

Se configura Kibana para su acceso a través de HTTPS en la interfaz VPN del honeywall y el puerto por defecto en la instalación:

Acceso Kibana: https://10.8.0.1:5601

Configuración en ANEXO A: página 112.

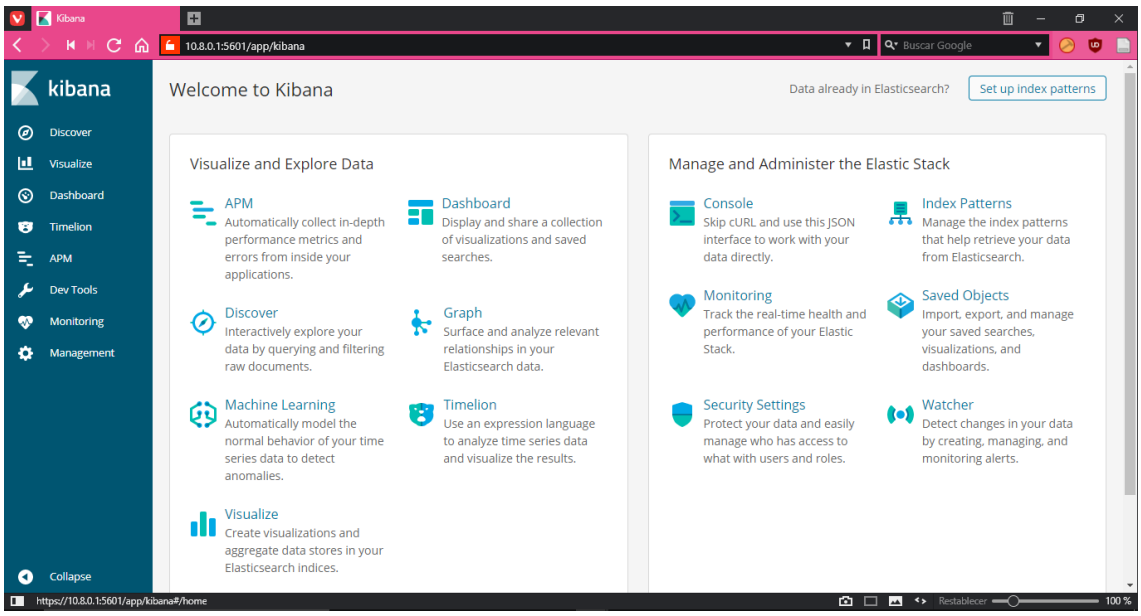

Figura 17 Página principal de Kibana

Se deben configurar en Kibana los índices de clasificación de datos establecidos con anterioridad para que Kibana sea capaz de realizar búsquedas con dichos datos. La configuración de los índices se realiza en Management > Kibana > Index Patterns.

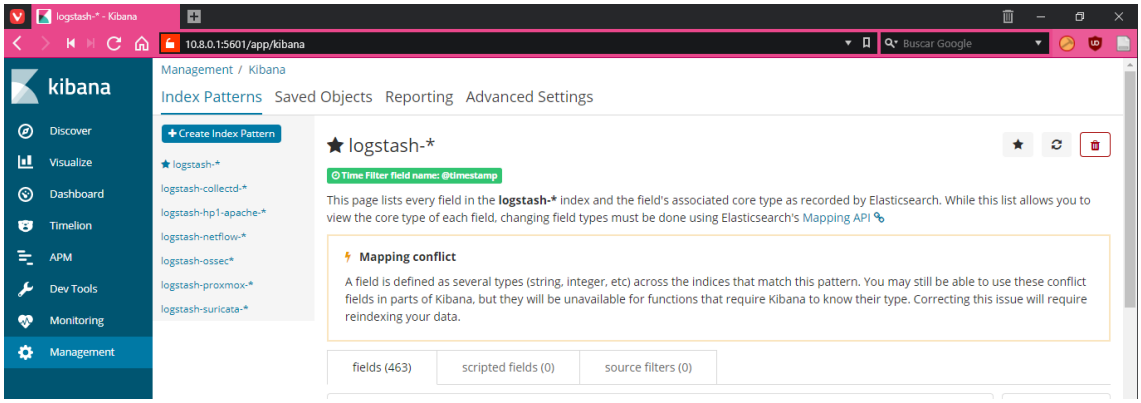

Figura 18 Índices de datos creados en Kibana

Una vez configurados los índices, en la sección Discover, Kibana muestra todos los datos pertenecientes a los índices configurados que está recogiendo Elasticsearch. Por ejemplo, se pueden observar los datos de estadísticas de red del puente de red del honeywall pertenecientes al índice: logstash-netflow-{fecha}

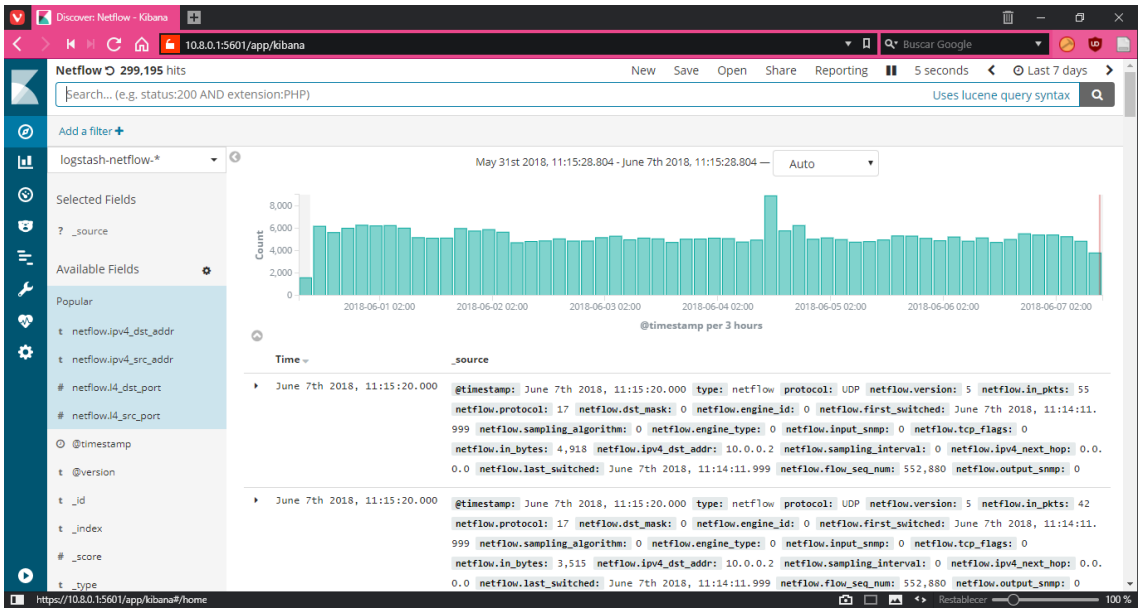

Figura 19 Datos de Elasticsearch en Kibana

Así mismo, se puede observar el formato JSON de entrada de los datos en Elasticsearch con un ejemplo del registro en una conexión en la honeynet

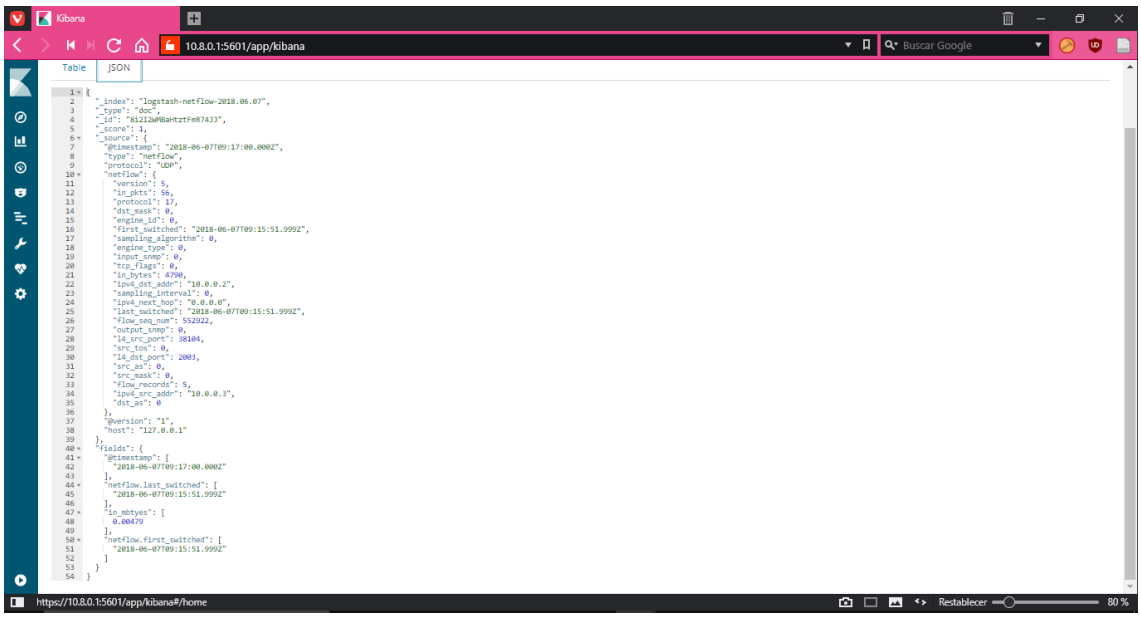

Figura 20 Registro JSON de Elasticsearch en Kibana

Cada uno de estos campos de datos de cada uno de los registros de datos de todos los índices configurados en Kibana son los que posteriormente se utilizan para la realización de búsquedas y la generación de gráficas de datos.

La visualización de datos en Kibana se realiza a través de gráficas y cuadros de resultados de búsquedas, por lo que se deben configurar todas las gráfica y búsquedas que se deseen en el apartado Visualize. Por el momento, se configuran las siguientes gráficas para los siguientes tipos de datos (se añadirán gráficas conforme se los diferentes honeypots y sus servicios):

9.5.2.1 *Estadísticas de uso del honeywall*

- Nombre: HWStat - CPU: datos sobre el porcentaje de uso de los diferentes estados de ejecución de toda la CPU del honeywall en el tiempo.

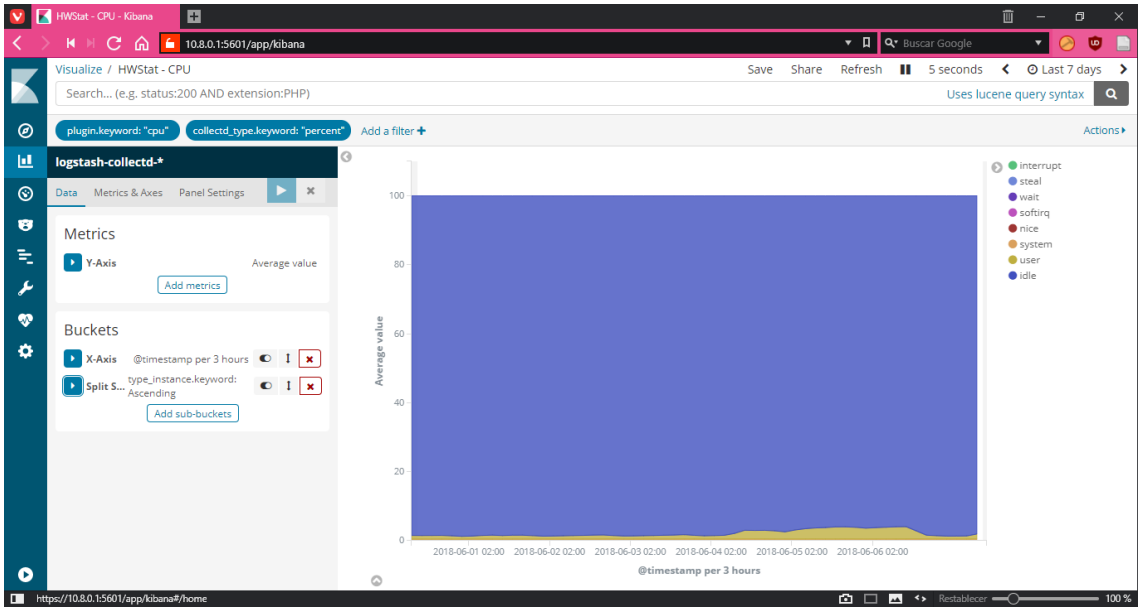

Figura 21 Gráfica HWStat – CPU

- Nombre: HWStat – RAM: estadísticas de uso de la memoria RAM del honeywall en el tiempo.

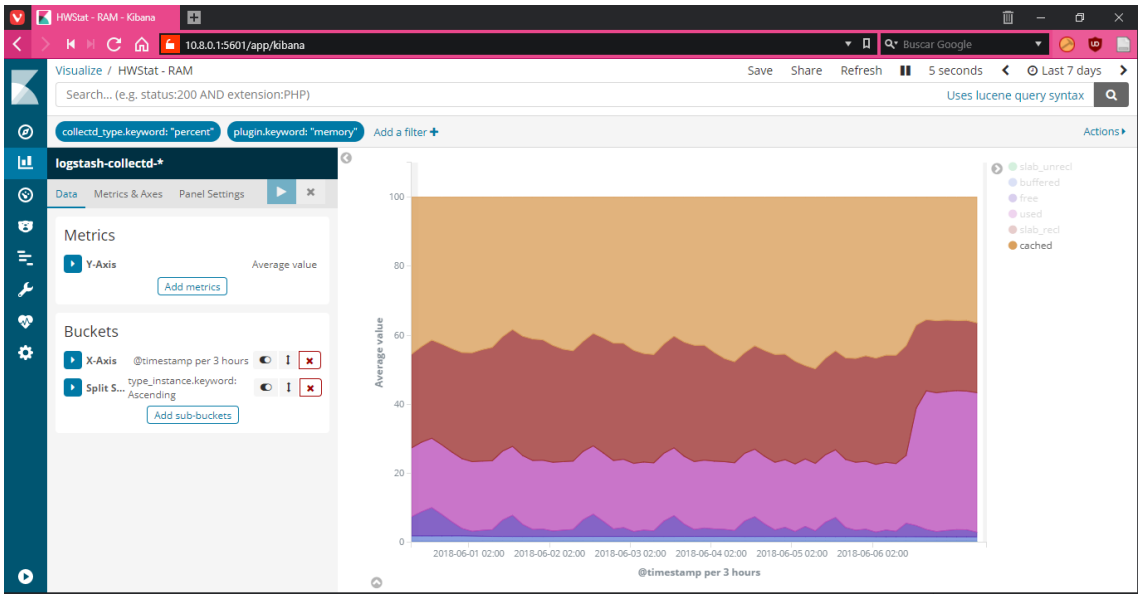

Figura 22 Gráfica HWStat – RAM

9.5.2.2 *Estadísticas de red de la honeynet*

- Nombre: Netflow – Bridge – MB: histórico de los Mb de datos transferidos por el puente de red del honeywall.

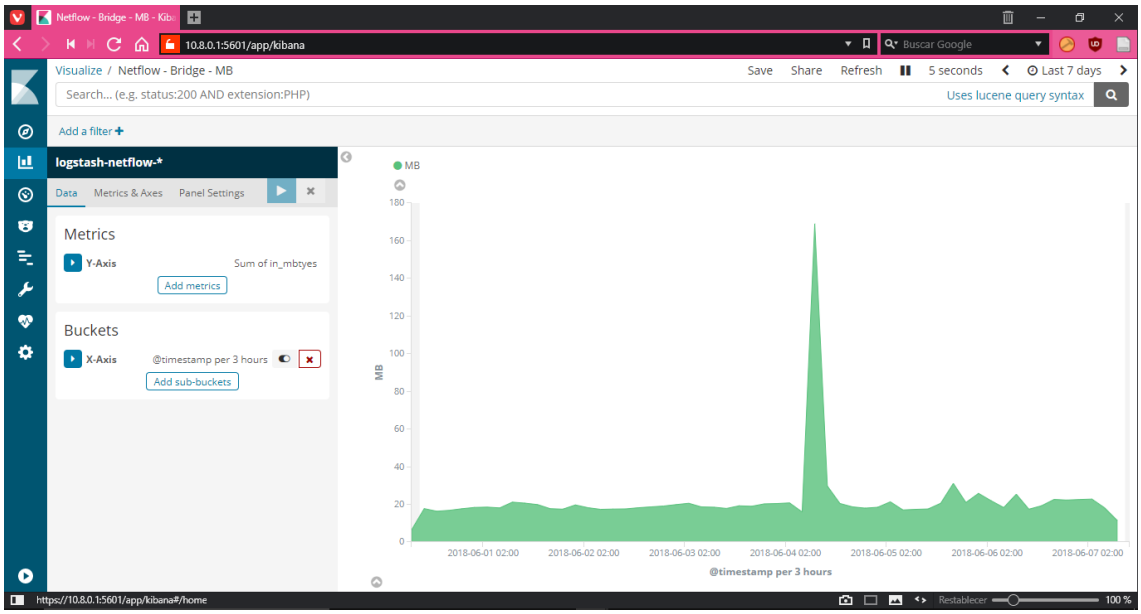

Figura 23 Gráfica Netflow - Bridge – MB

- Nombre: Netflow – Sesiones: cuadro de información de sesiones establecidas en la honeynet.

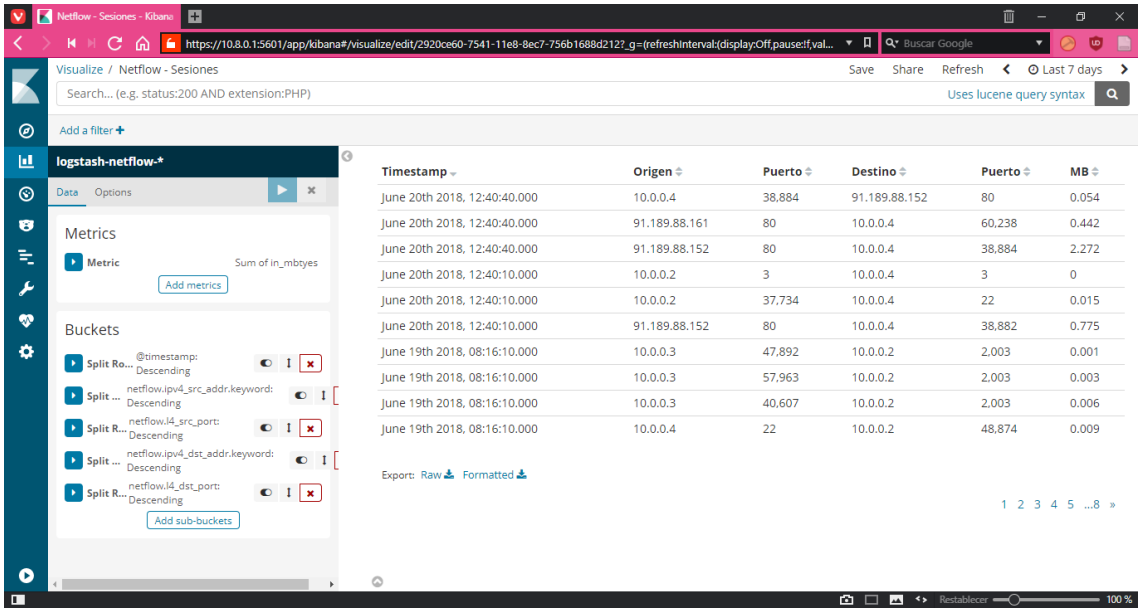

Figura 24 Gráfica Netflow - Sesiones

- Nombre: Netflow – Búsqueda: panel de búsqueda de datos de IPs origen/destino y puertos origen/destino.

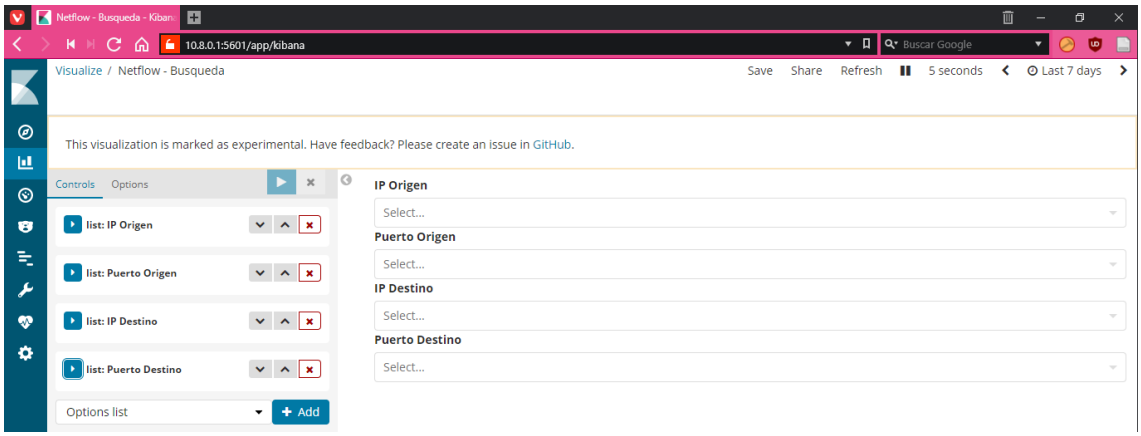

Figura 25 Panel de búsqueda Netflow – Búsqueda

9.5.2.3 Prevención de intrusiones en red

- Nombre: NIPS – Event type: histórico de los diferentes eventos que Suricata detecta en la honeynet.

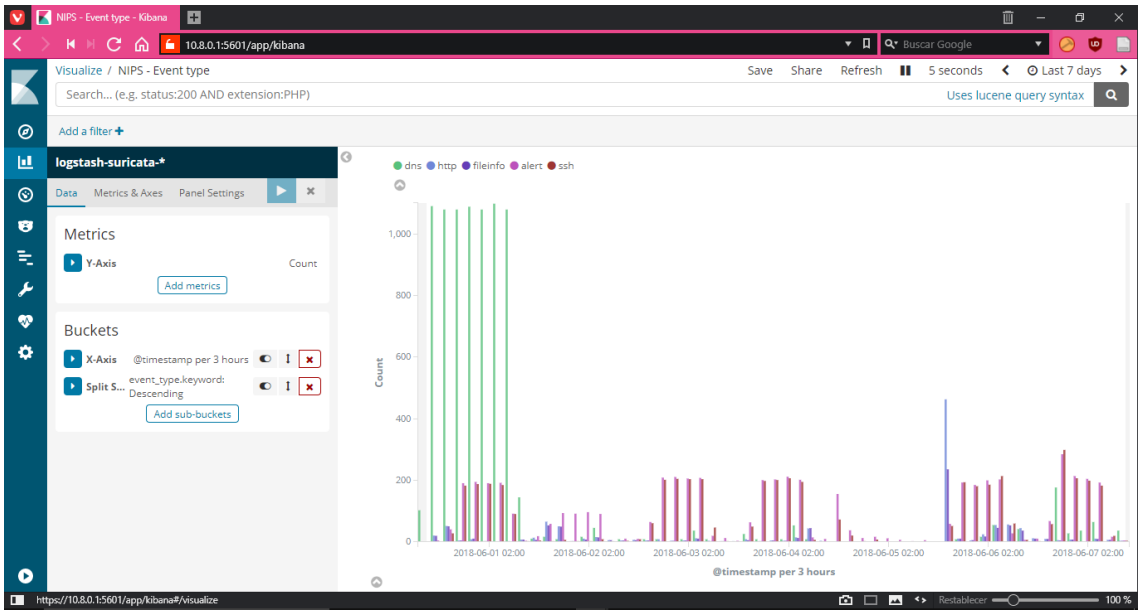

Figura 26 Gráfica NIPS - Event type

- Nombre: NIPS – AvsB: comparación del histórico de tráfico permitido y bloqueado por Suricata.

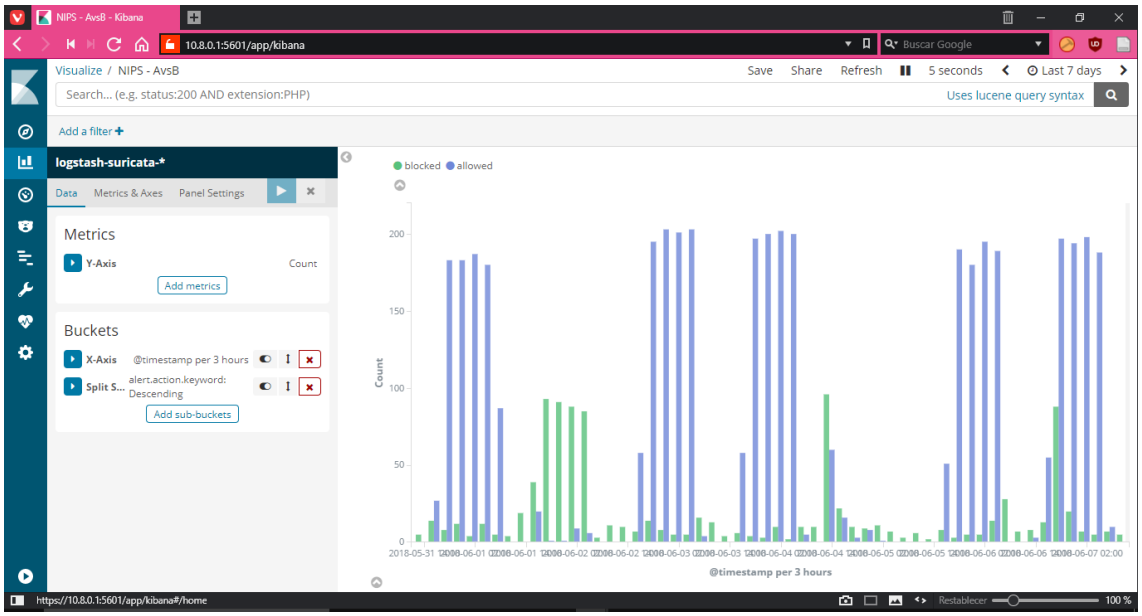

Figura 27 Gráfica NIPS – AvsB

- Nombre: NIPS – Top 5 signatures allowed: gráfica que muestra las reglas que más veces ha utilizado Suricata para alertar de tráfico malicioso y cuya acción es permitir el tráfico.

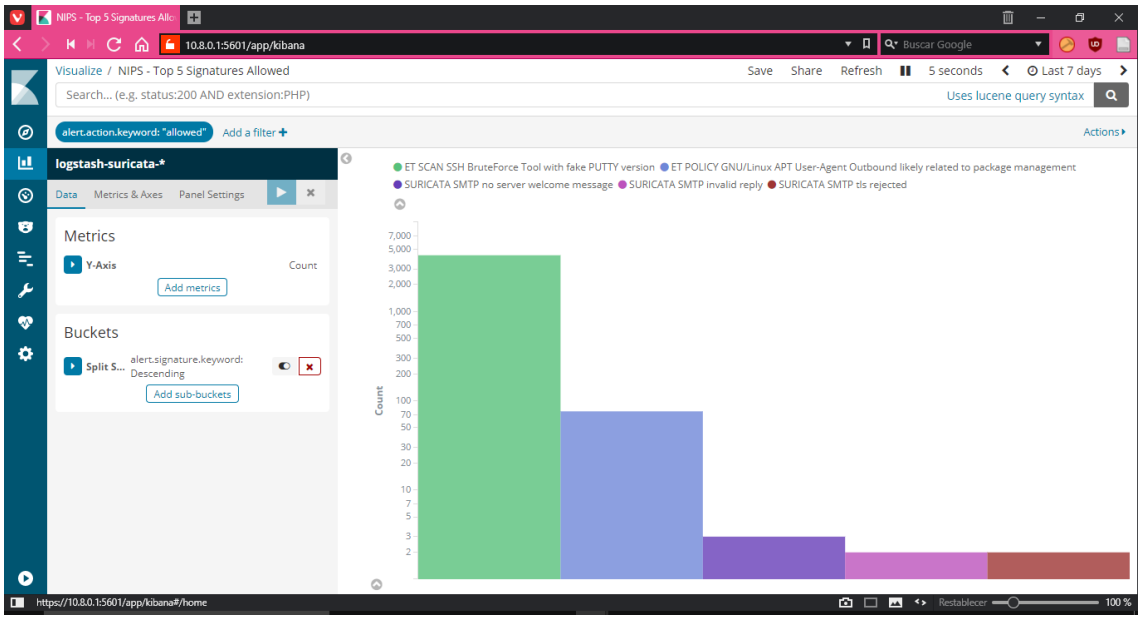

Figura 28 Gráfica NIPS – Top 5 signatures allowed

- Nombre: NIPS – Top 5 signatures blocked: gráfica que muestra las reglas que más veces ha utilizado Suricata para alertar de tráfico malicioso y cuya acción es bloquear el tráfico.

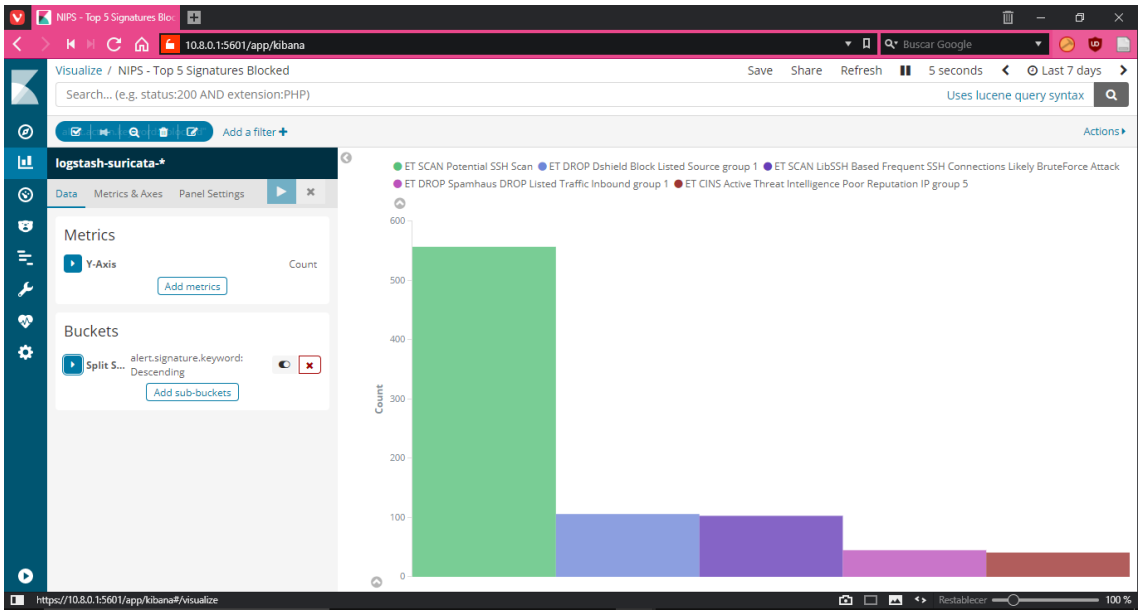

Figura 29 NIPS – Top 5 signatures blocked

- Nombre: NIPS - Top 10 IP-Ports Allowed: gráfica de tarta que muestra la relación de las IPs y puertos relacionados con el tráfico alertado y permitido por Suricata.

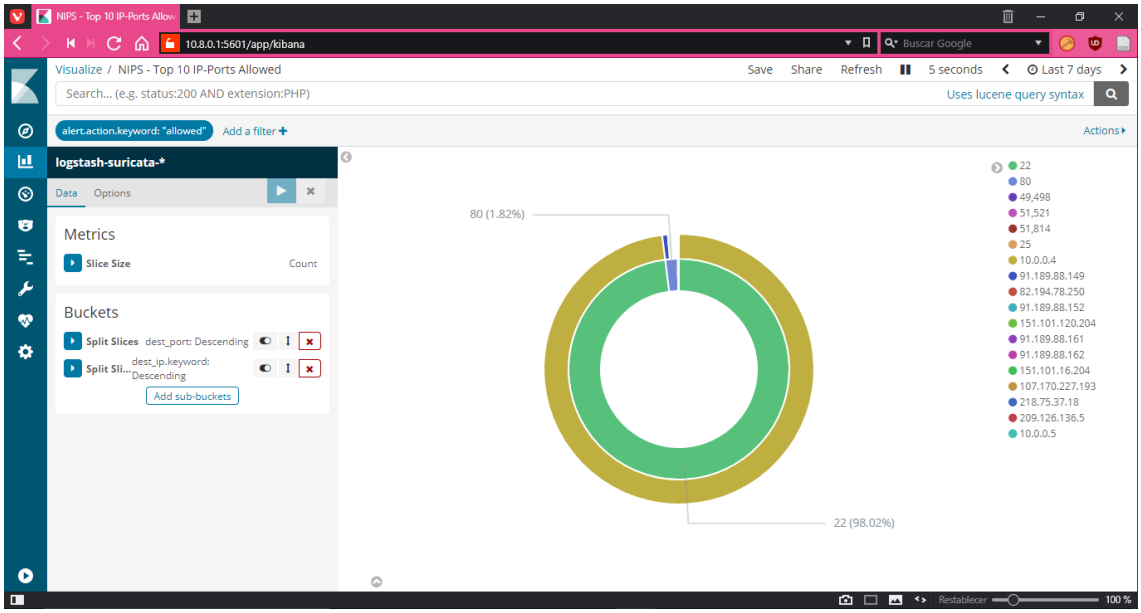

Figura 30 Gráfica NIPS - Top 10 IP-Ports Allowed

- Nombre NIPS - Top 10 IP-Ports Blocked: gráfica de tarta que muestra la relación de las IPs y puertos relacionados con el tráfico alertado y bloqueado por Suricata.

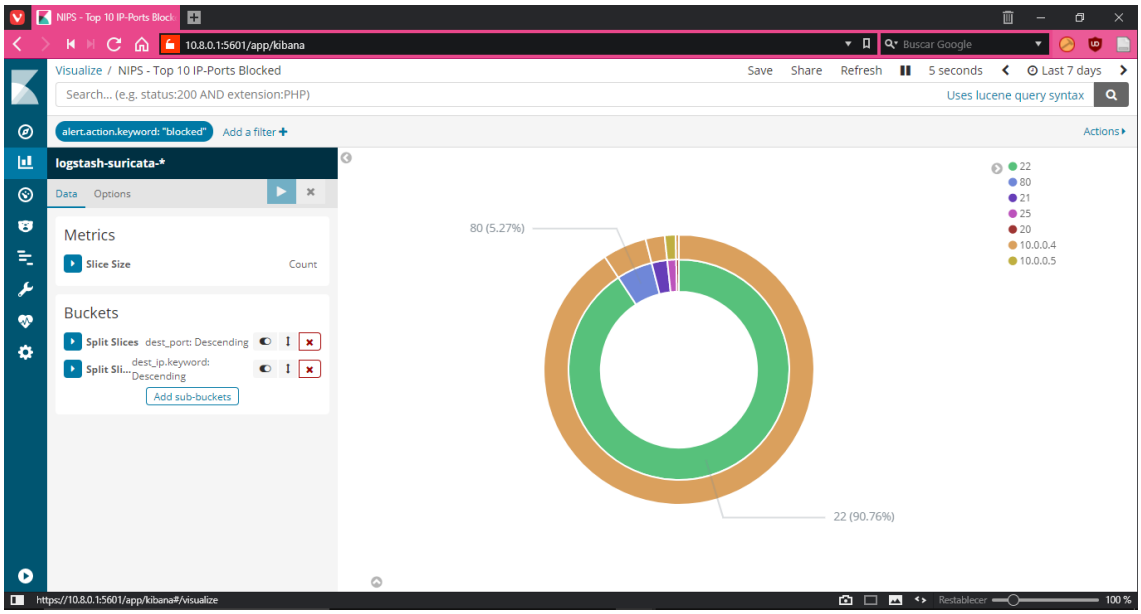

Figura 31 Gráfica NIPS - Top 10 IP-Ports Blocked

El gráfico de integración de las herramientas de análisis de datos y administración de la honeynet es el siguiente:

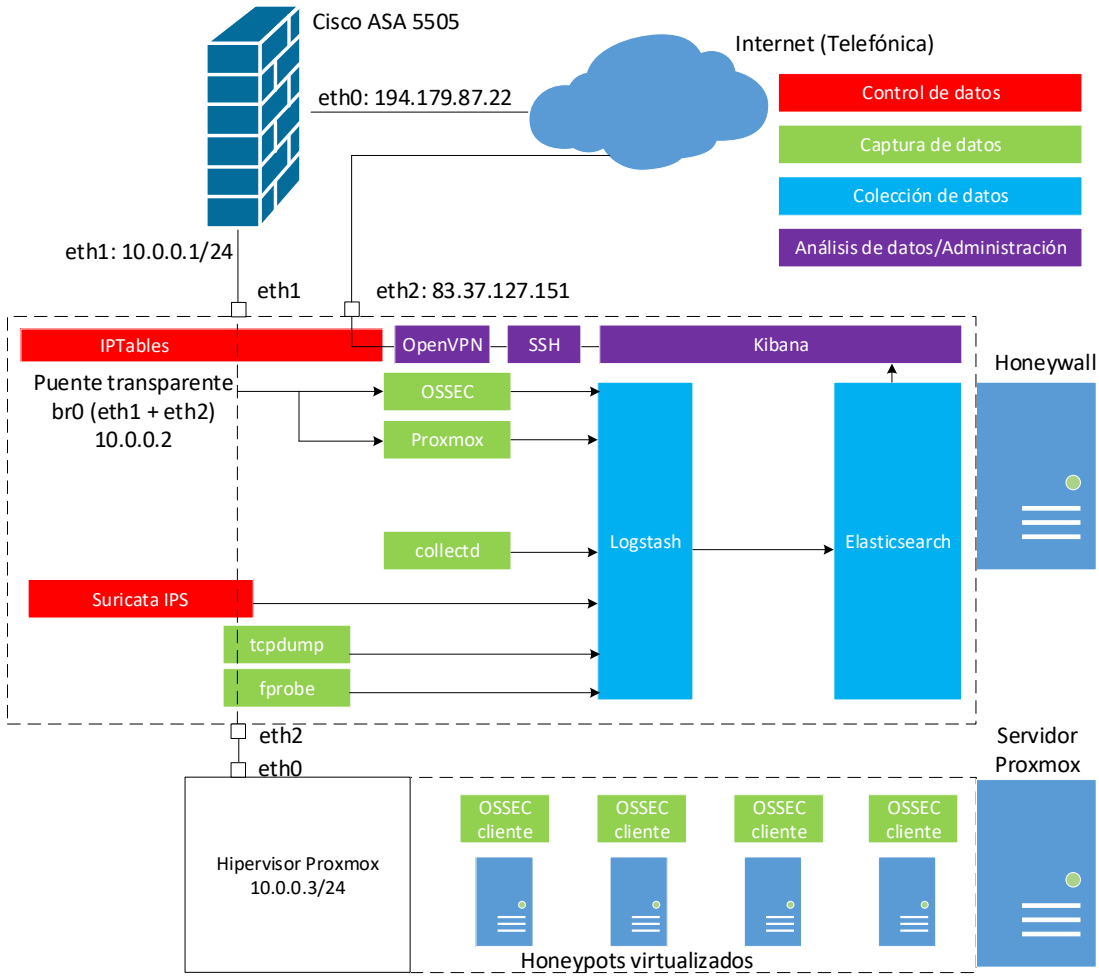

Figura 32 Herramientas de análisis de datos y administración en la honeynet

Se puede observar como el servidor OpenVPN para el acceso a través del túnel VPN escucha en la interfaz de administración eth2 con la dirección IP pública 83.37.127.151, así como el servidor SSH y la interfaz web Kibana son accesibles sólo a través del túnel VPN.

9.6 Software de los honeypots

9.6.1 Nivel de interacción

Para este proyecto, se opta por configurar todos los servicios de los diferentes honeypots como servicios de alta interacción, es decir, servicios reales que ofrecen una completa interactividad con el atacante, para disimular al máximo la existencia de la honeynet e incrementar al máximo el valor de todos los datos que se recojan en la red.

9.6.2 Honeypots desplegados

9.6.2.1 Honeypot HP1

El primer honeypot virtualizado que se despliega en la honeynet es tiene las siguientes características básicas:

- ID Proxmox: 100.
- Sistema Operativo: Ubuntu Server 16.04.3 de 64 bits.
- CPU: 1 CPU de 4 núcleos.
- Memoria RAM: 4 GB.
- Disco: 32 GB.
- Dirección IP: 10.0.0.4/24

Configuración en ANEXO A: página 112.

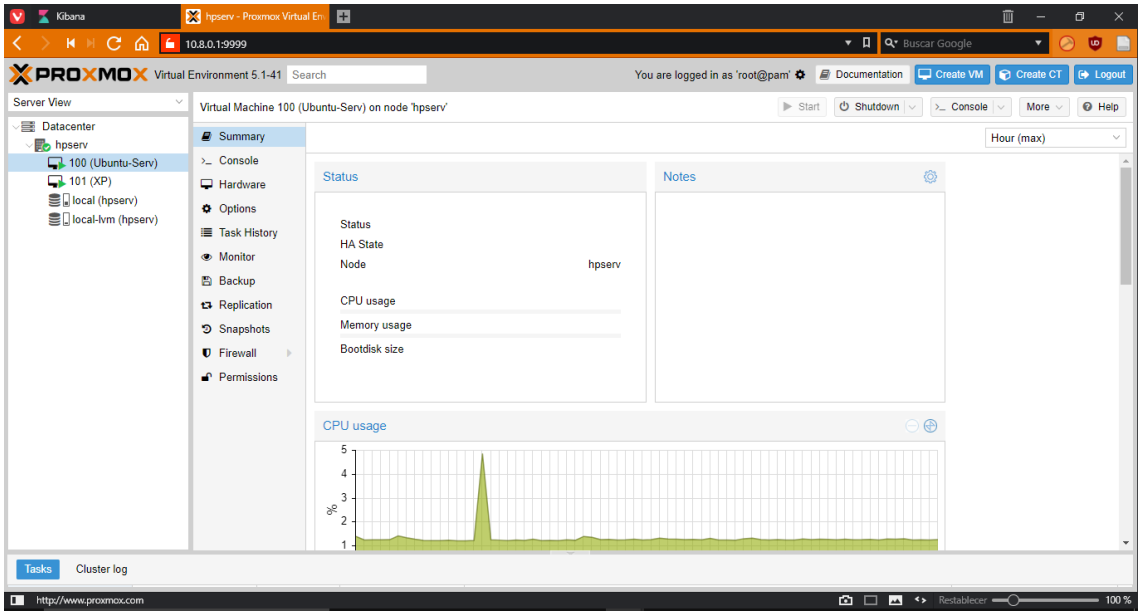

Figura 33 Máquina virtual HP1 en el servidor Proxmox

El primer servicio desplegado en el honeypot es un servidor web Apache, versión 2.4.18, cuya configuración se deja tal cual se establece en la instalación por defecto del servidor. Las vulnerabilidades de dicha versión de Apache se encuentran disponibles en la base de datos de CVE [10]. El objetivo de este servicio no es ofrecer un servidor web antiguo y vulnerable

hacia Internet, sino la posibilidad de estudiar los diferentes escaneos automáticos de servicios web que se realizan y las direcciones IPs origen de dichos escaneos. Se debe configurar el cortafuegos frontera de la honeynet CISCO ASA 5505 para que redirija mediante NAT estático toda las peticiones al puerto HTTP/80 de la dirección pública al honeypot en cuestión:

http://194.179.87.22:80 -> NAT -> http://10.0.0.4:80

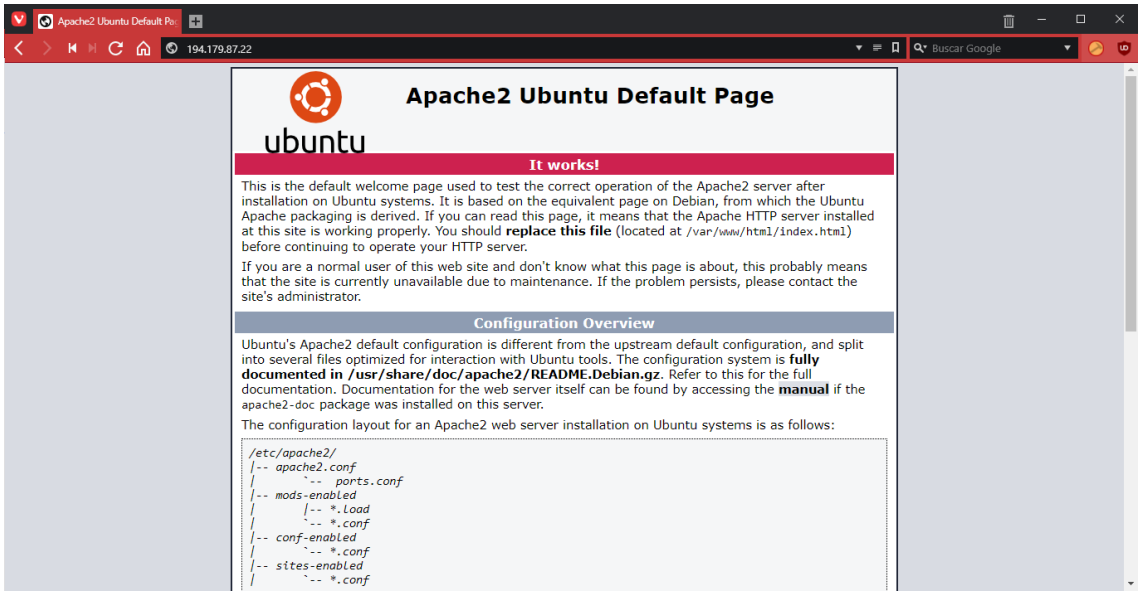

Figura 34 Honeypot servidor web

El segundo servicio desplegado en el honeypot es un servidor FTP configurado mediante el software vsftpd, versión 3.0.3, cuya instalación se modifica para el acceso mediante un usuario/contraseña fácilmente adivinable como test/test. No existen vulnerabilidades conocidas para ese software y esa versión en concreto. El objetivo de este servicio es ofrecer una instalación de un servidor FTP con un sistema de autenticación vulnerable para observar las técnicas de ataque en el acceso a un servidor FTP y su uso posterior. Se debe configurar el cortafuegos frontera de la honeynet CISCO ASA 5505 para que redirija mediante NAT estático toda las peticiones al puerto FTP/21 de la dirección pública al honeypot en cuestión:

ftp://194.179.87.22:21 -> NAT -> ftp://10.0.0.4:21

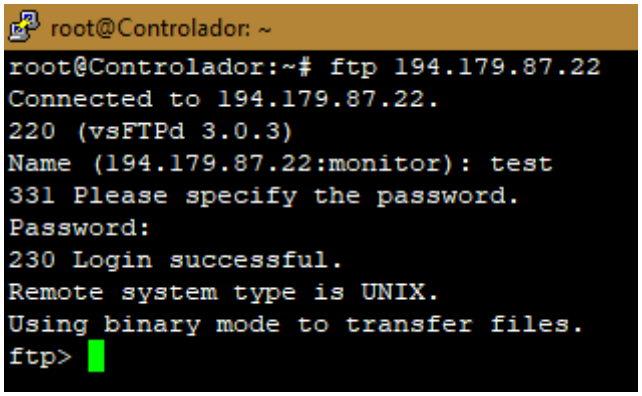

Figura 35 Honeypot servidor FTP

El tercer y último servicio desplegado en el honeypot es un servidor SSH configurado mediante el servidor OpenSSH, versión 7.2p2. El servidor SSH se configura para el acceso mediante los siguientes usuarios y contraseñas:

- test/test
- dev-ubuntu/dev-ubuntu

Dichas cuentas de usuario tienen privilegios mínimos en el honeypot y no se permite ni el acceso ni el login como administrador mediante la herramienta sudo y similares. El objetivo del servicio es observar los ataques de fuerza bruta que se realizan para acceder al servidor y capturar todas las acciones que un atacante lleva a cabo una vez dentro del honeypot. La vulnerabilidades conocidas del servidor SSH en la versión 7.2p2 se encuentran disponibles en la base de datos CVE [11]. Se debe configurar el cortafuegos frontera de la honeynet CISCO ASA 5505 para que redirija mediante NAT estático toda las peticiones al puerto SSH/22 de la dirección pública al honeypot en cuestión:

ssh 194.179.87.22:22 -> NAT -> 10.0.0.4:22

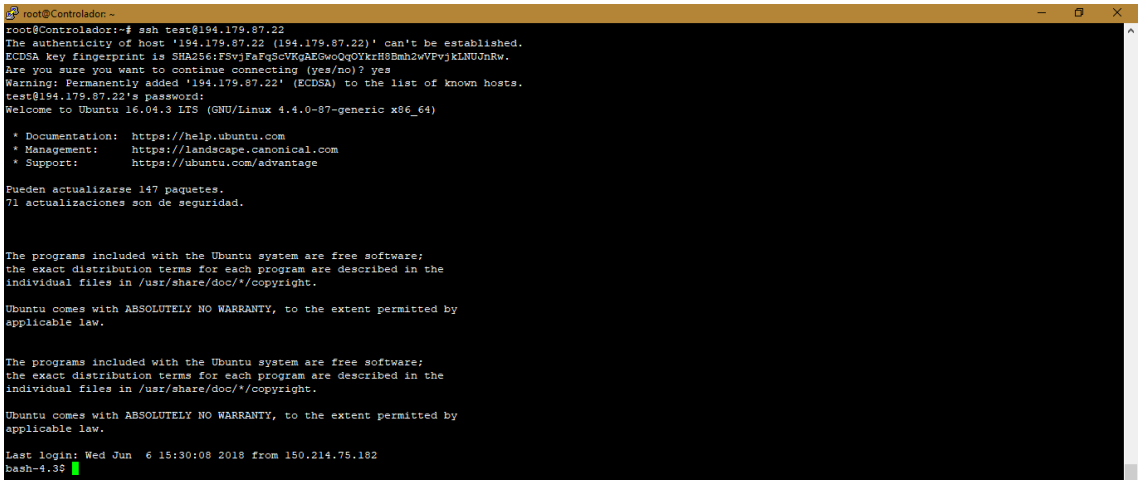

Figura 36 Honeypot servidor SSH

Para la correcta captura de datos de los servicios instalados en este honeypot, se configura un HIDS OSSEC en modo cliente, añadiendo las reglas especificadas en el Anexo A: Configuraciones en el servidor para la detección de:

- Intento de usuario incorrecto en SSH.
- Intento fallido de inicio de sesión SSH con usuario correcto.
- Intento de usuario incorrecto en FTP.
- Operaciones con ficheros en el servidor FTP (subida, descarga y borrado).

Se debe añadir a Logstash, en el honeywall, la configuración para leer los logs de acceso al servidor web del honeypot HP1 a través del fichero /var/log/hp1/apache/access.log, así como el índice de clasificación de datos de Logstash: logstash-hp1-apache-{fecha}.

El gráfico de interacción del honeypot HP1 con la honeynet es el siguiente

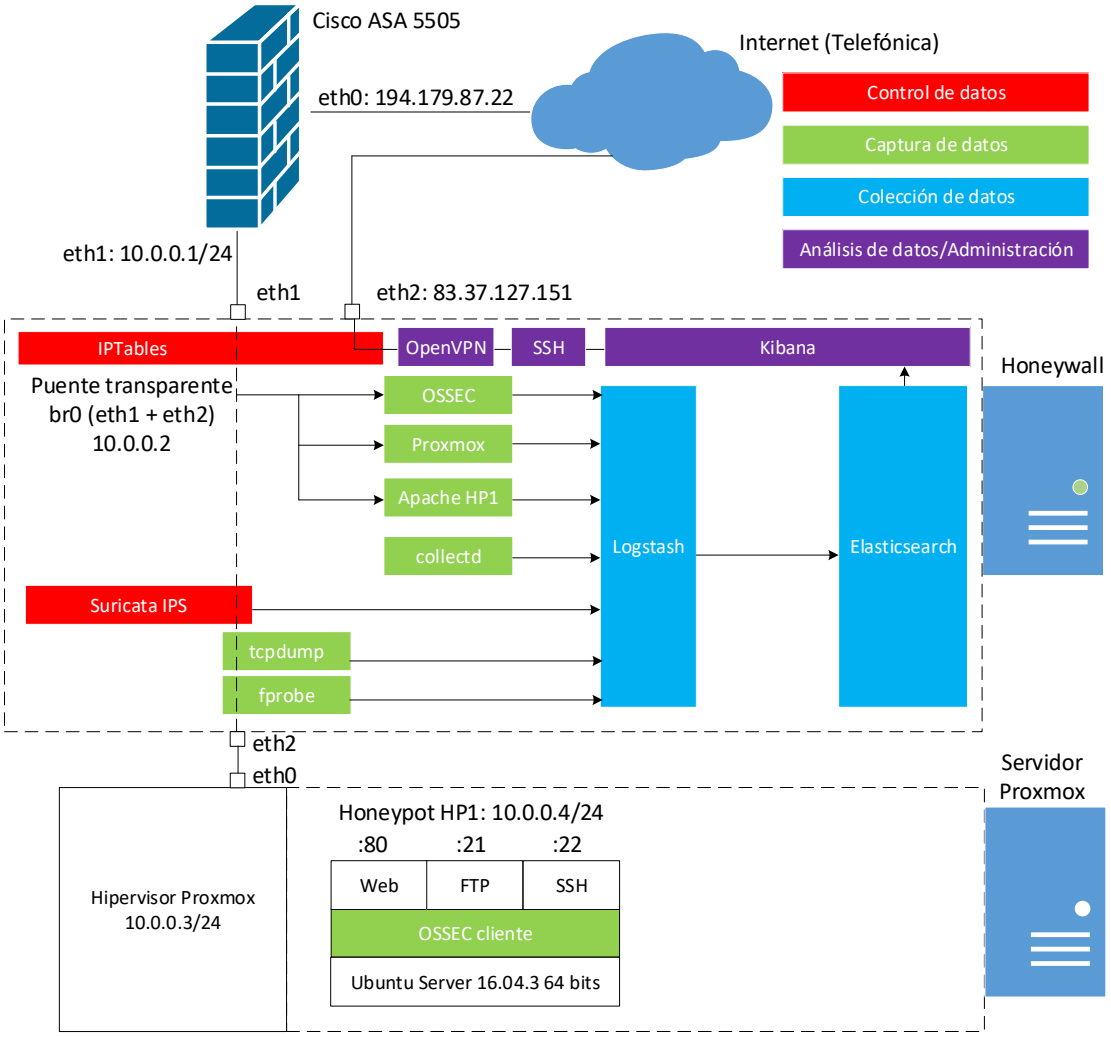

Figura 37 Honeypot HP1 en la honeynet

Así mismo, se añaden a Kibana las diferentes gráficas de datos capturados relacionados con el honeypot HP1 en cuestión:

- Nombre: HP1 – Stat – CPU: histórico del uso de CPU del honeypot HP1

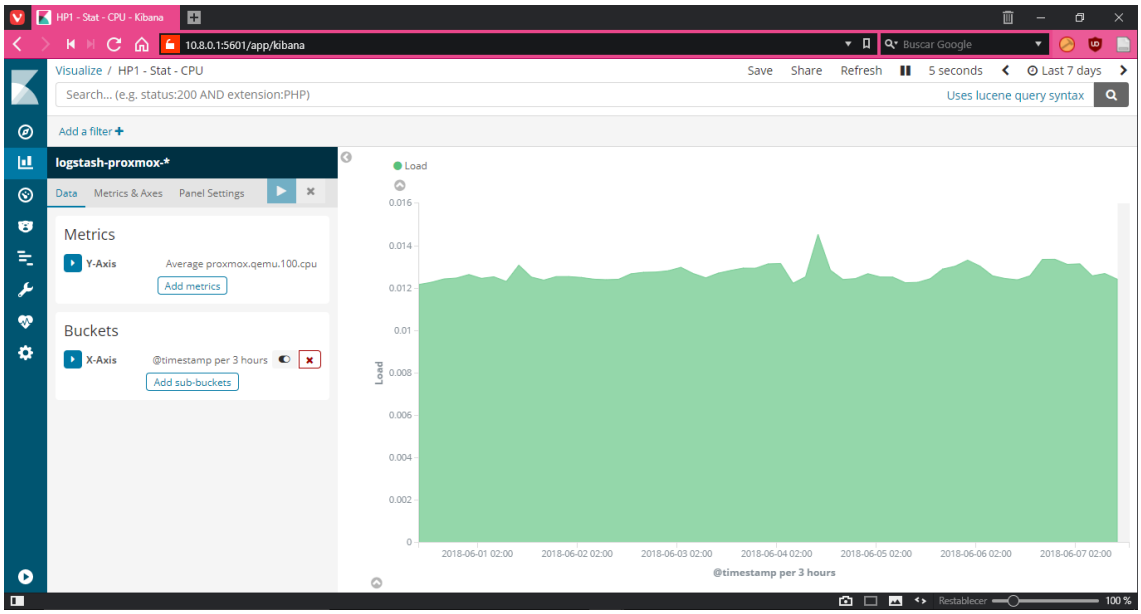

Figura 38 Gráfica HP1 – Stat – CPU

- Nombre: HP1 – Stat – Mem: histórico del uso de memoria RAM del honeypot HP1

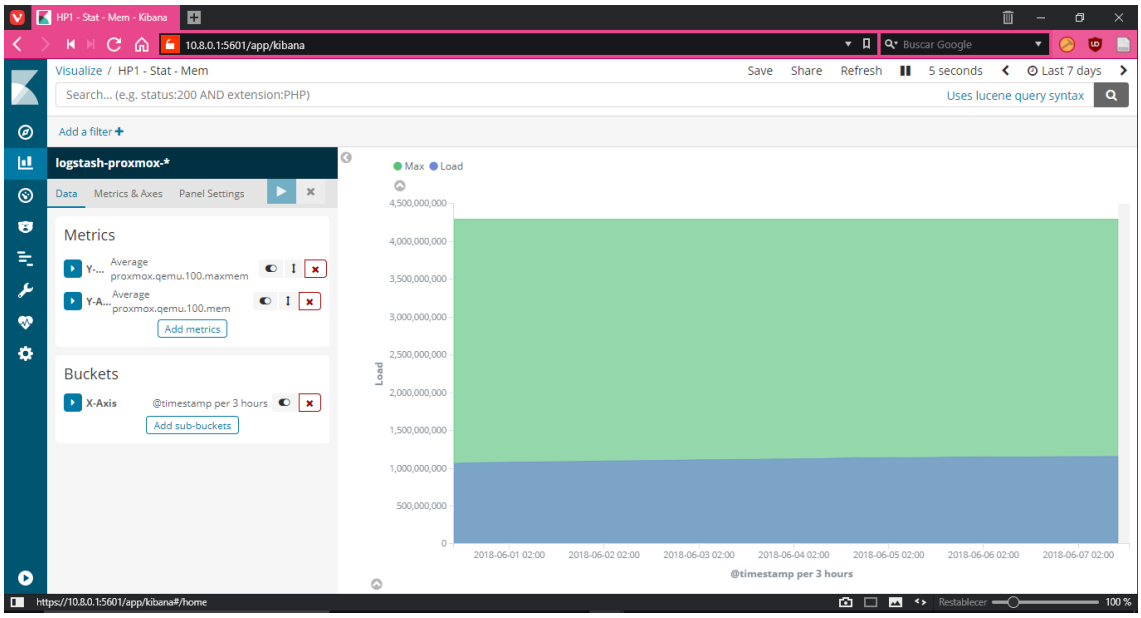

Figura 39 Gráfica HP1 – Stat – Mem

- Nombre: Netflow - Bridge - Stat - HP1: histórico del tráfico de red relacionado con el honeypot HP1

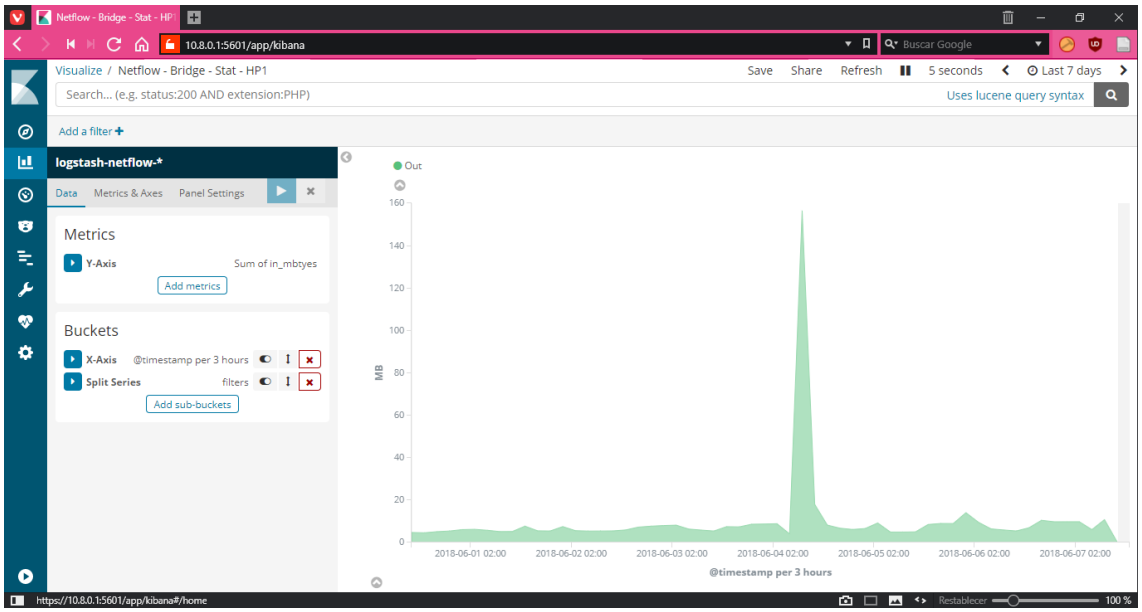

Figura 40 Gráfica Netflow - Bridge - Stat - HP1

- Nombre: Netflow - HP1 – In: gráfico de tarta de relaciones de IPs origen y puertos del honeypot HP1 para conexiones entrantes, ordenadas por Mb transferidos.

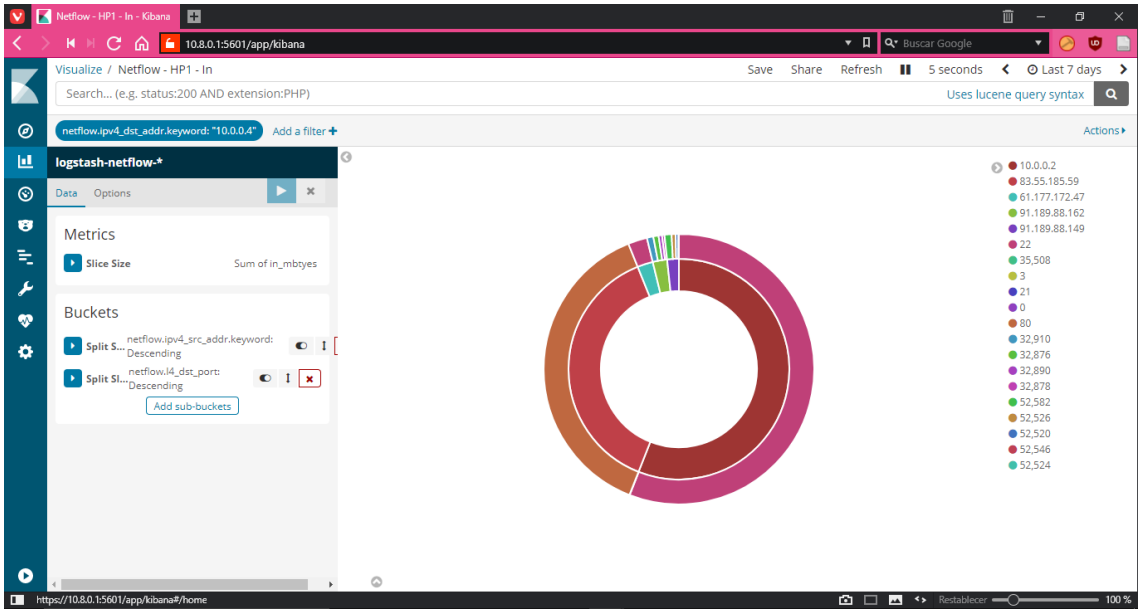

Figura 41 Gráfica Netflow - HP1 – In

- Nombre: Netflow - HP1 – Out: gráfico de tarta de relaciones de IPs destino y puertos del honeypot HP1 para conexiones salientes, ordenadas por Mb transferidos.

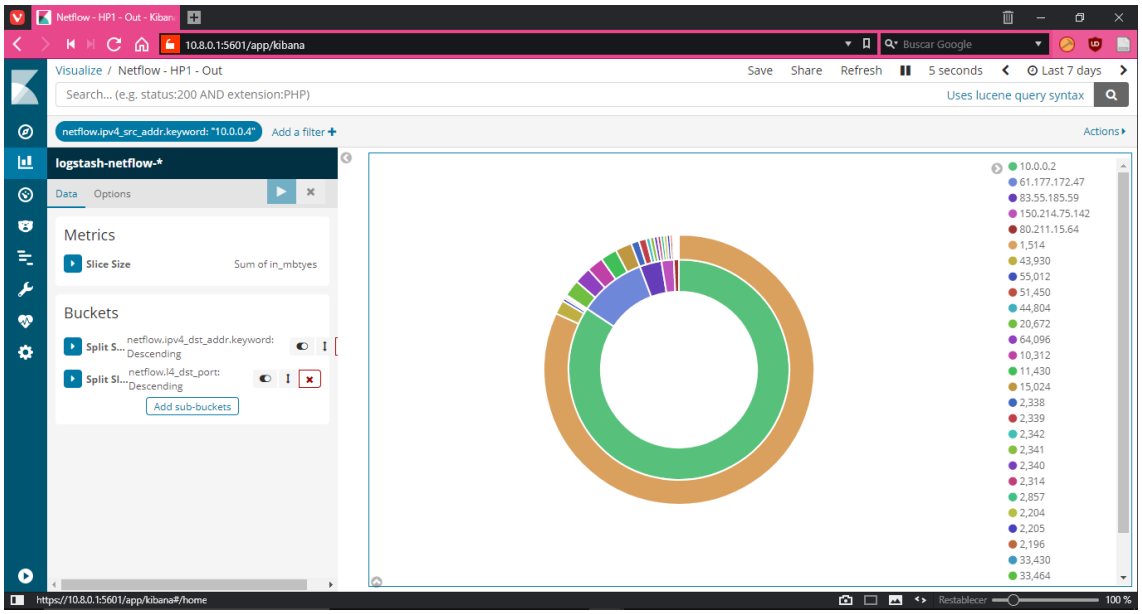

Figura 42 Gráfica Netflow - HP1 – Out

- Nombre: HP1 – Apache – GETvsPOST: diferencia entre número de peticiones GET y POST que se realizan al servidor.

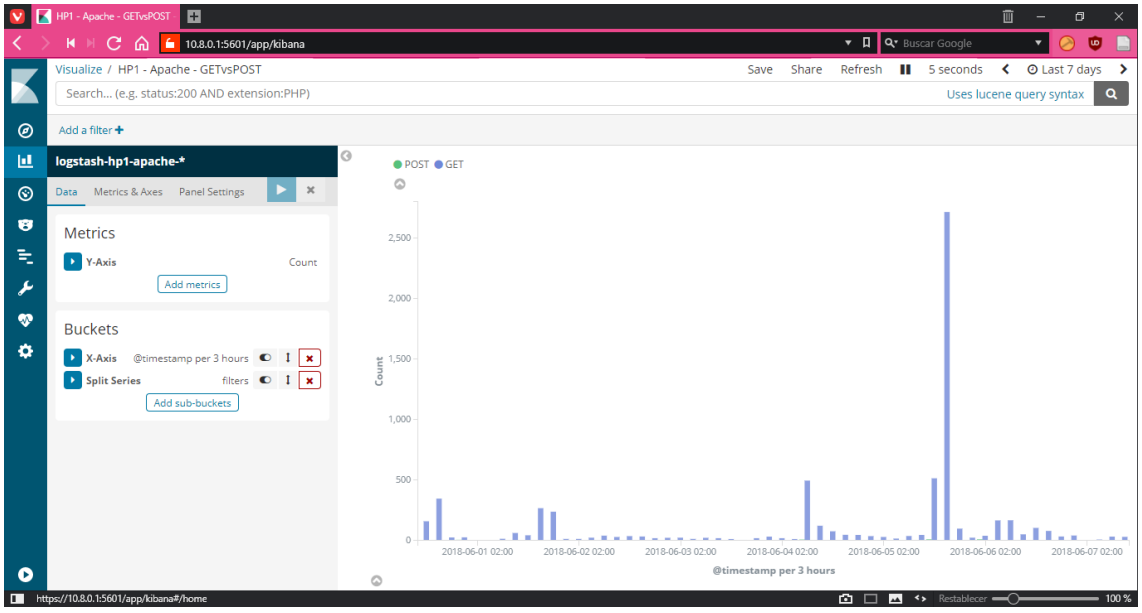

Figura 43 Gráfica HP1 – Apache – GETvsPOST

- Nombre: HP1 - Apache - Top 10 IPs: gráfica de las IPs origen que más peticiones realizan al servidor web del honeypot HP1

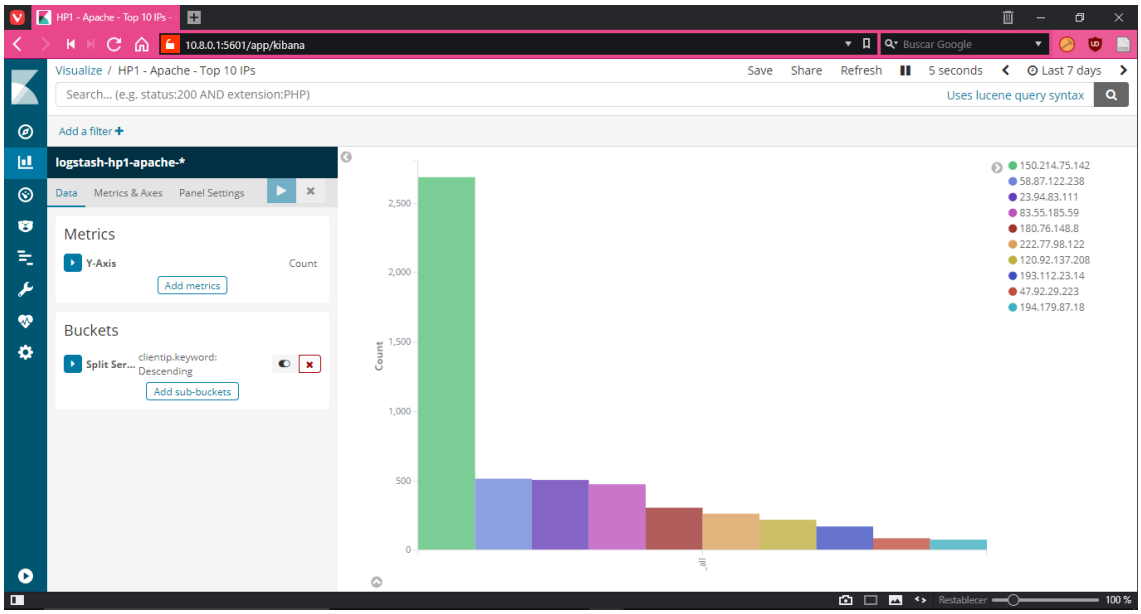

Figura 44 Gráfica HP1 - Apache - Top 10 IPs

- Nombre: HP1 - Apache - Top 10 Path: gráfica de las URLs que más se solicitan en el servidor web del honeypot HP1

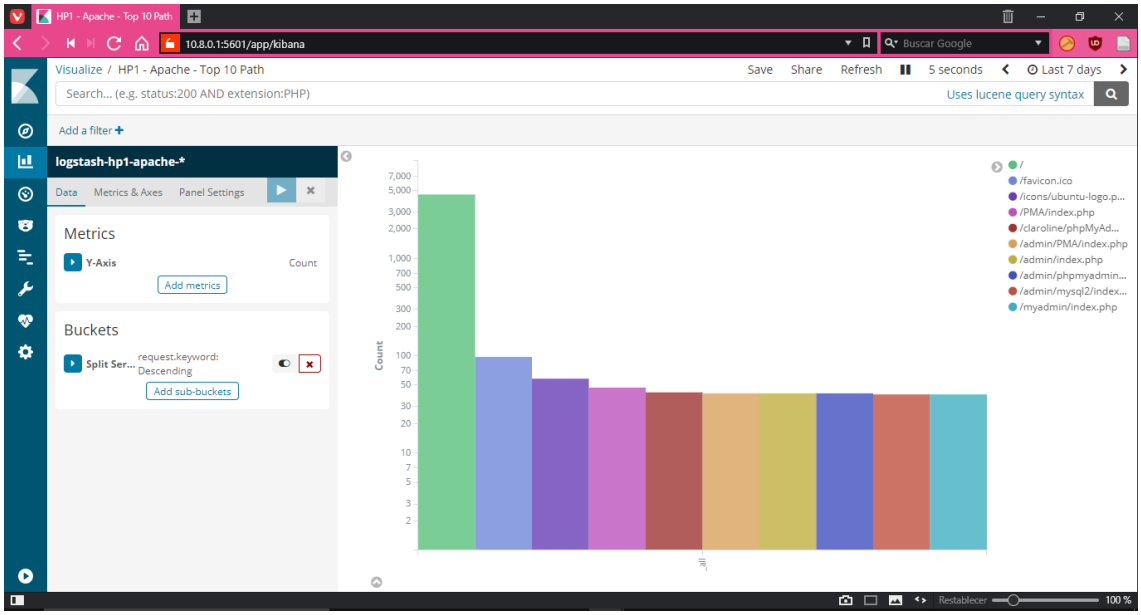

Figura 45 Gráfica HP1 - Apache - Top 10 Path

- Nombre: HP1 - Apache – UA: gráfica de los User-Agents que más peticiones realizan al servidor web del honeypot HP1

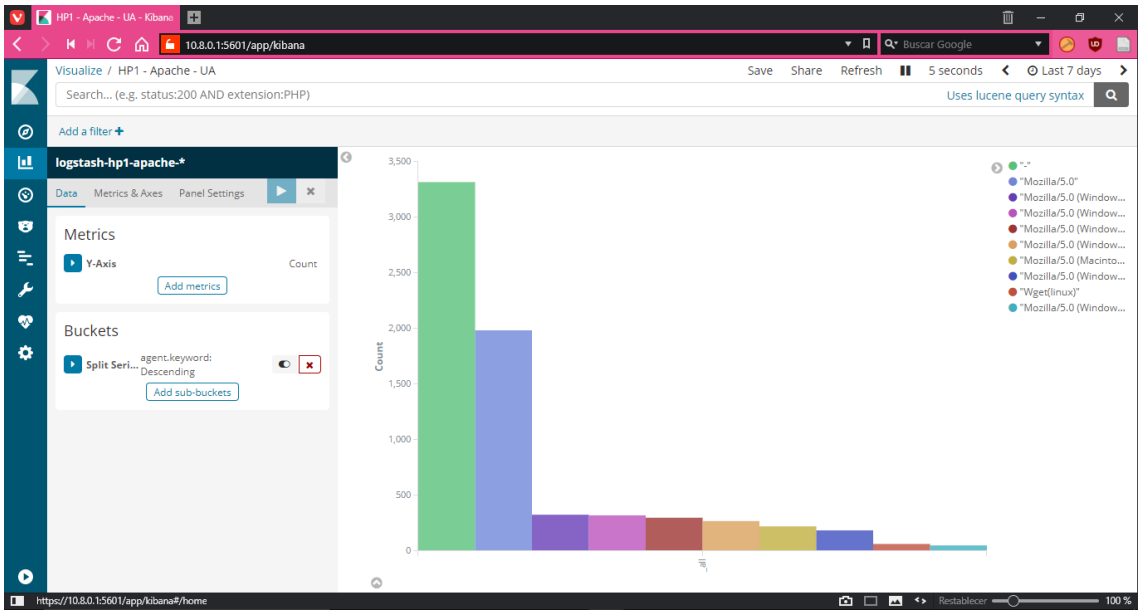

Figura 46 Gráfica HP1 - Apache – UA

- Nombre: HP1 - FTP - Login OK. Conteo de inicios de sesión correctos en el servidor FTP del honeypot HP1

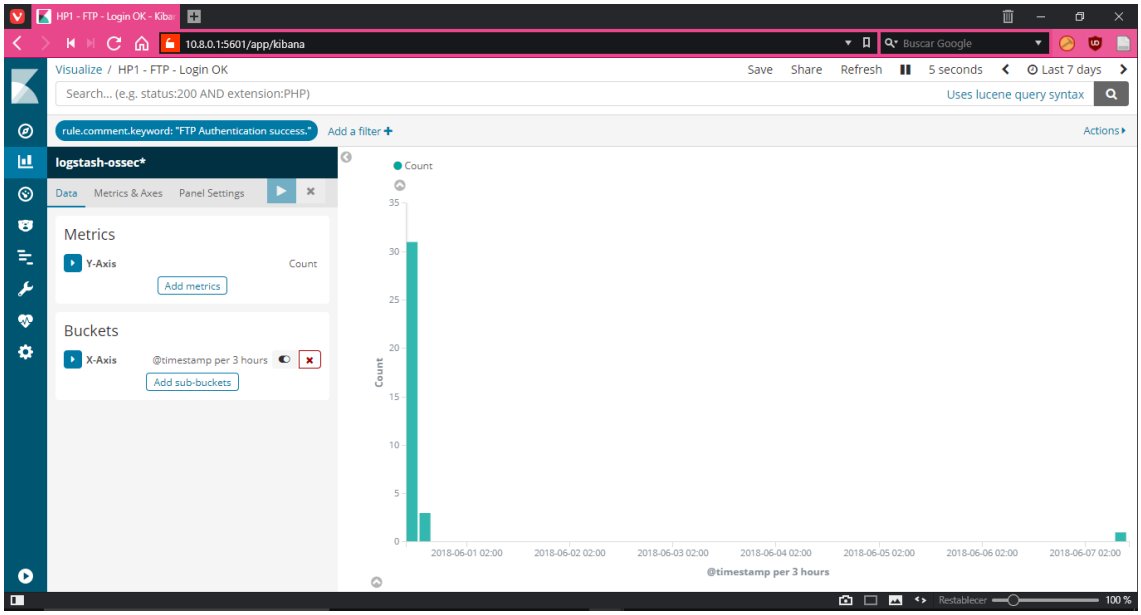

Figura 47 Gráfica HP1 - FTP - Login OK

- Nombre: HP1 - FTP - Login Failed. Nombres de usuario incorrectos que mas se intentan en el servidor FTP del honeypot HP1

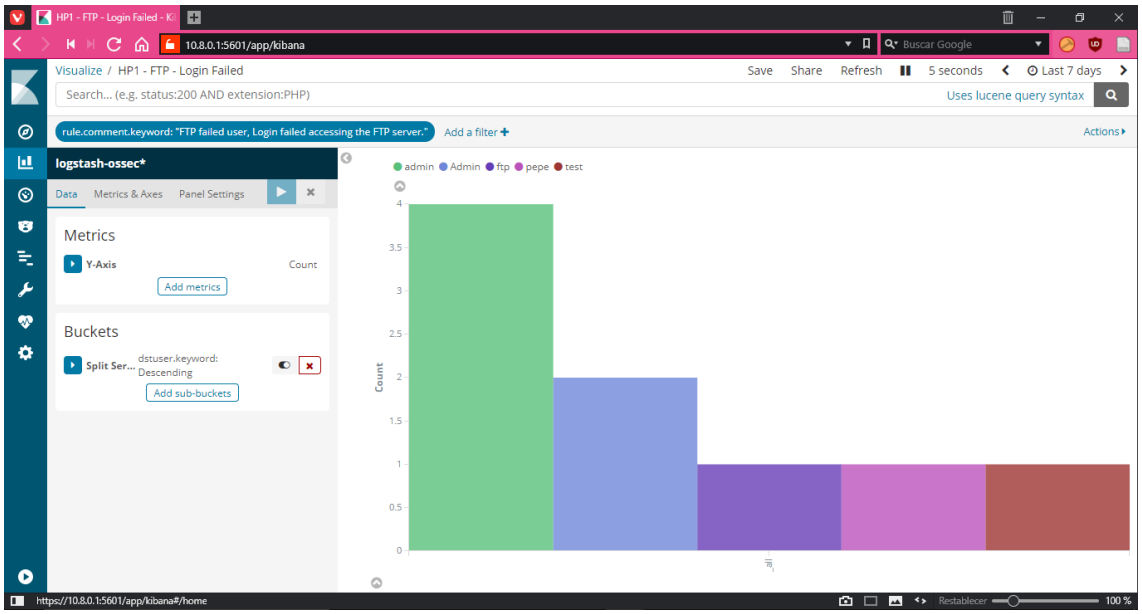

Figura 48 Gráfica HP1 - FTP - Login Failed

- Nombre: HP1 - FTP - Failed – IP: relaciones de usuarios incorrectos por diferentes direcciones IPs, ordenadas por el número de intentos

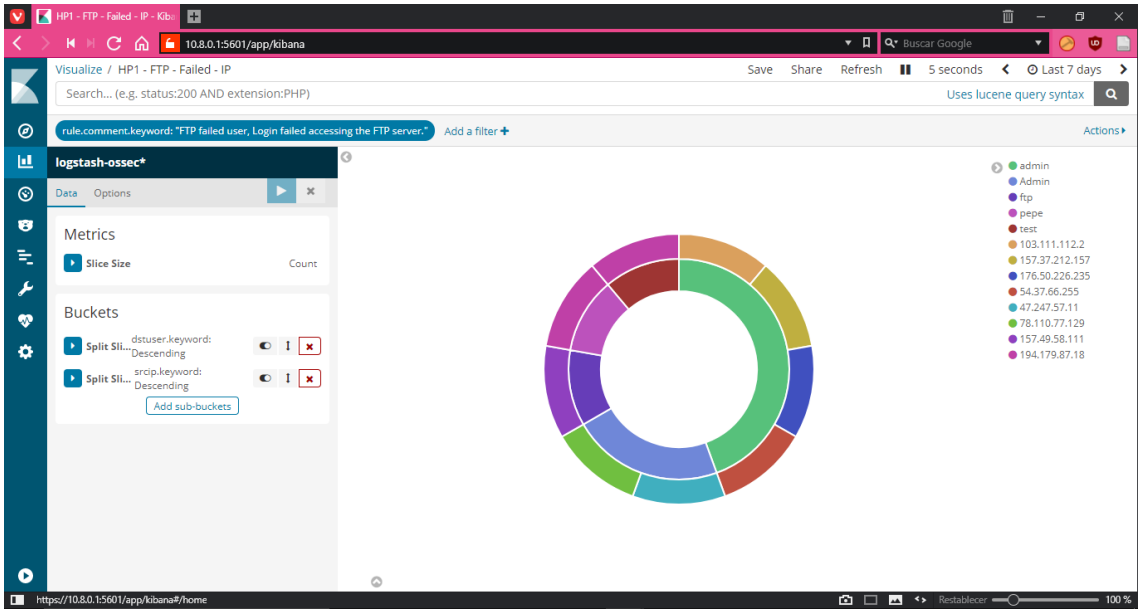

Figura 49 Gráfica HP1 - FTP - Failed – IP

- Nombre: HP1 - FTP - Files DUD: histórico del número de operaciones sobre ficheros realizadas en el servidor FTP del honeypot HP1

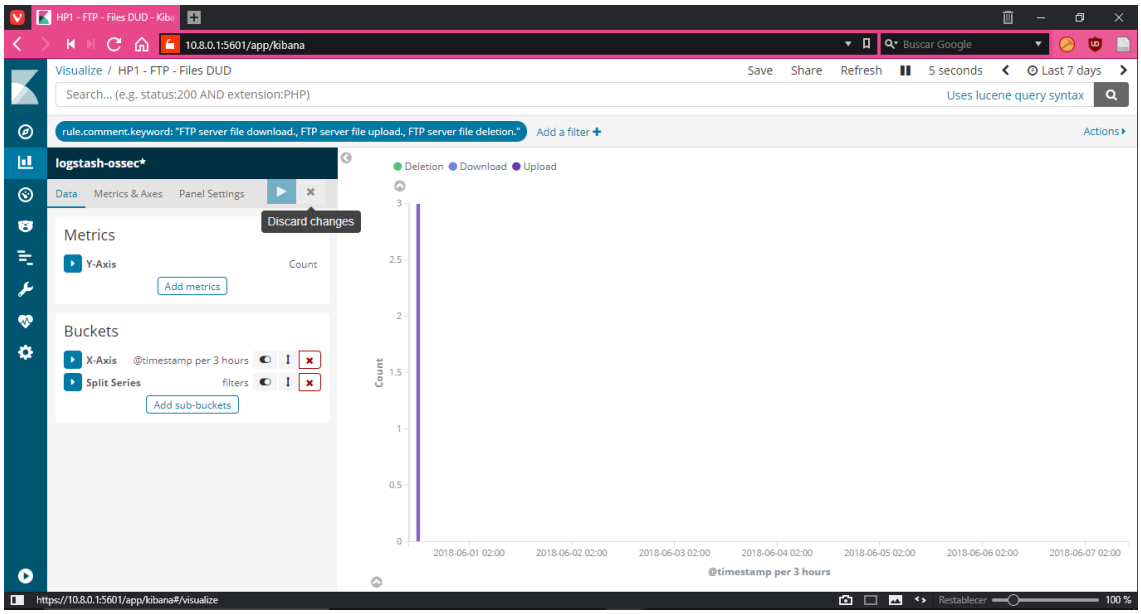

Figura 50 Gráfica HP1 - FTP - Files DUD

- Nombre: HP1 - SSH - Success root and user: histórico del número de inicio de sesión en el servidor SSH del honeypot HP1

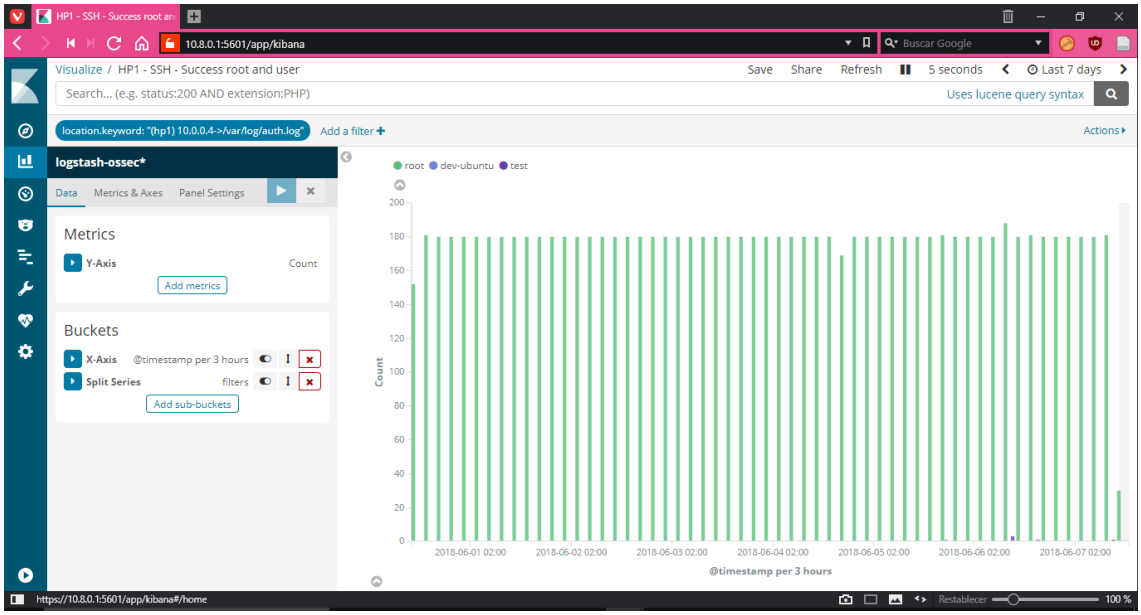

Figura 51 Gráfica HP1 - SSH - Success root and user

- Nombre: HP1 - SSH - Top failed users: usuarios fallidos más repetidos en el servidor SSH del honeypot HP1

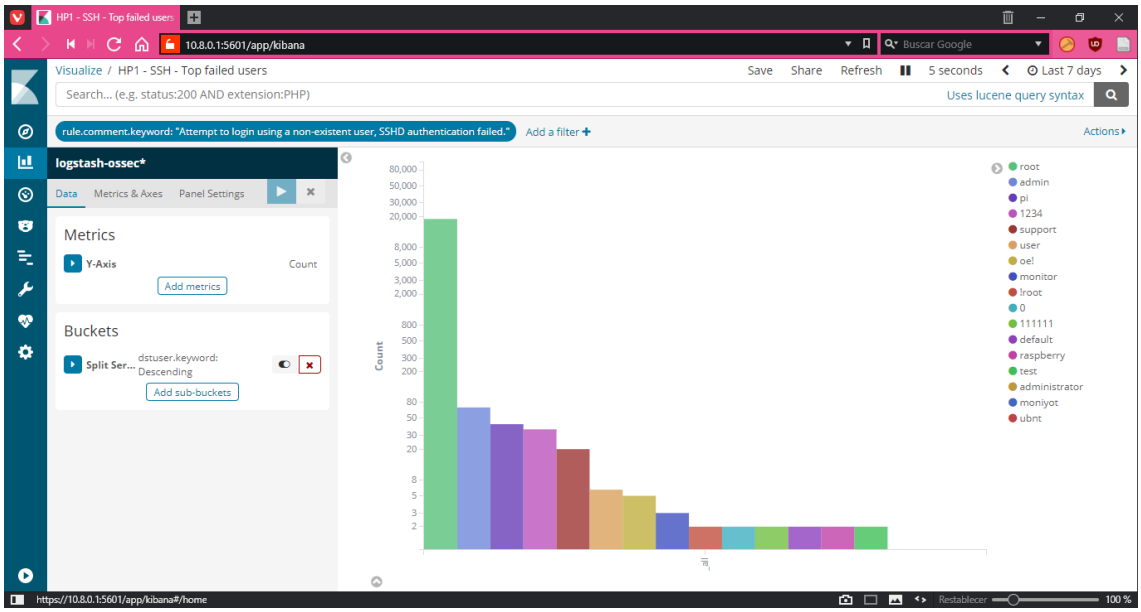

Figura 52 Gráfica HP1 - SSH - Top failed users

- Nombre: HP1 - SSH - Failed – IP: relación de usuarios fallidos más repetidos con las direcciones IPs origen de los intentos al servidor SSH del honeypot HP1

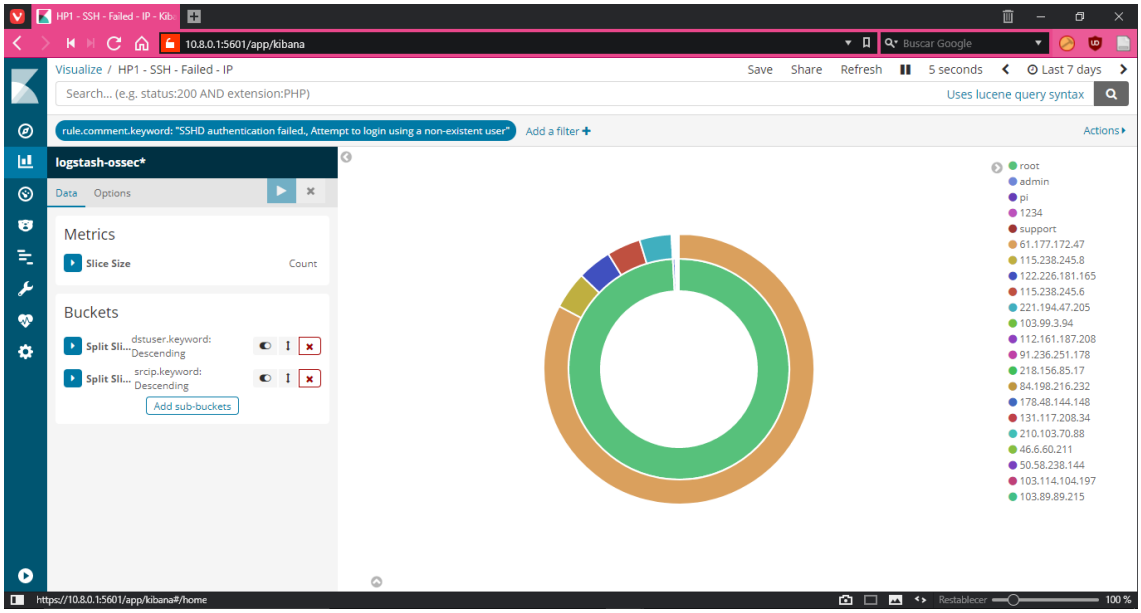

Figura 53 Gráfica HP1 - SSH - Failed – IP

9.6.2.2 Honeypot HP2

El segundo honeypot virtualizado que se despliega en la honeynet es tiene las siguientes características básicas:

- ID Proxmox: 101.
- Sistema Operativo: Windows XP Service Pack 3.
- CPU: 1 CPU de 2 núcleos.
- Memoria RAM: 2 GB.
- Disco: 32 GB.
- Dirección IP: 10.0.0.5/24

Configuración en ANEXO A: página 114.

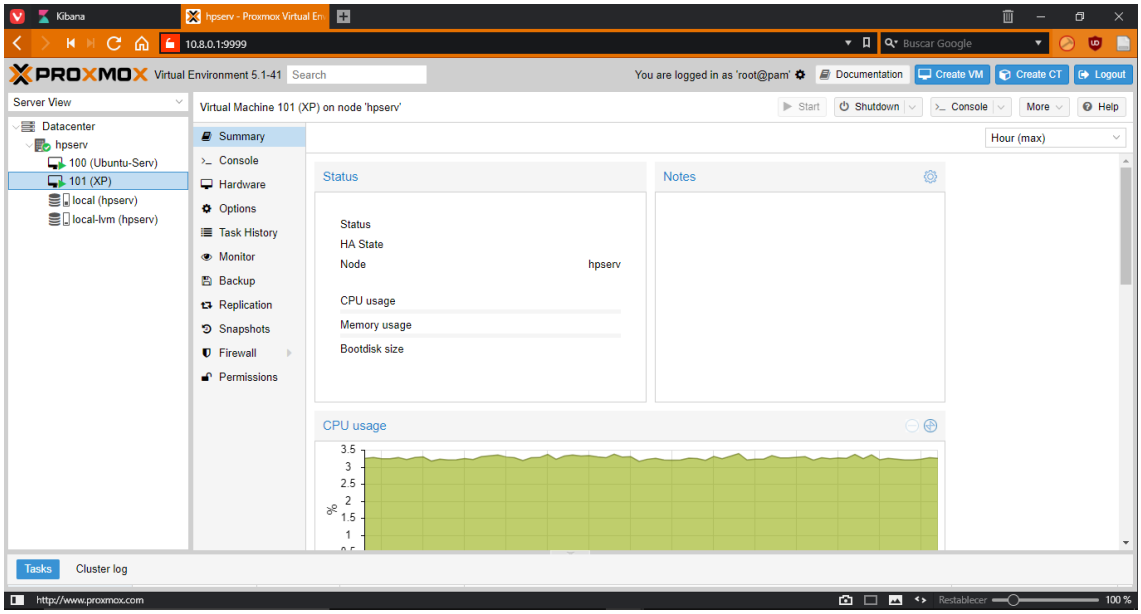

Figura 54 Máquina virtual HP2 en el servidor Proxmox

El único servicio desplegado en el honeypot HP2 es un servidor SMTP, configurado mediante una instalación por defecto del servidor IIS, versión 6.0. Las vulnerabilidades conocidas de dicho servidor se encuentran disponibles en la base de datos CVE [12]. El objetivo de este servicio es ofrecer la posibilidad a la honeynet de capturar campañas de spam y phishing para prevenir sobre posibles ataques de ingeniería social en la red en producción. Si se capturan los mensajes de correo enviados al servidor SMTP, con las capturas de tráfico se puede reconstruir el mensaje, así como su contenido adjunto, posibilitando el análisis de posible malware. Se debe configurar el cortafuegos frontera de la honeynet CISCO ASA 5505 para que redirija mediante NAT estático toda las peticiones al puerto SMTP/25 de la dirección pública al honeypot en cuestión:

smtp 194.179.87.22:25 -> NAT -> 10.0.0.5:25

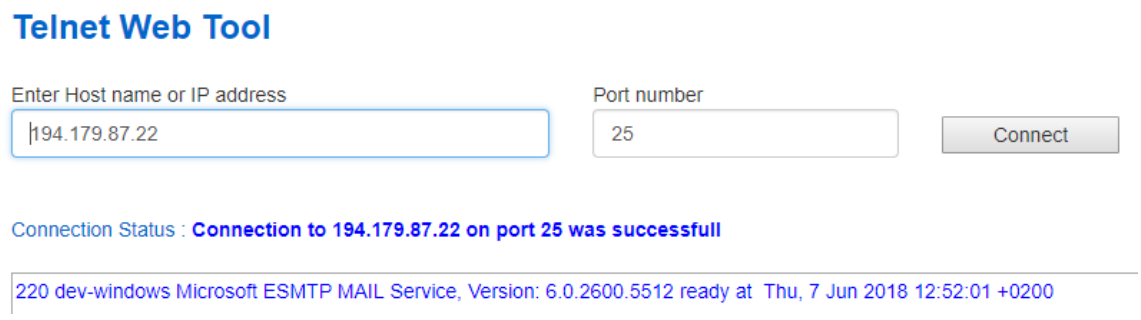

Figura 55 Acceso vía Telnet al servidor SMTP del honeypot HP2

En este honeypot no se configura un cliente HIDS OSSEC porque se considera que todos los datos que este honeypot puede ofrecer se pueden extraer con las herramientas de captura instaladas en el honeywall.

El gráfico de interacción del honeypot HP2 con el resto de los elementos de la honeynet es el siguiente:

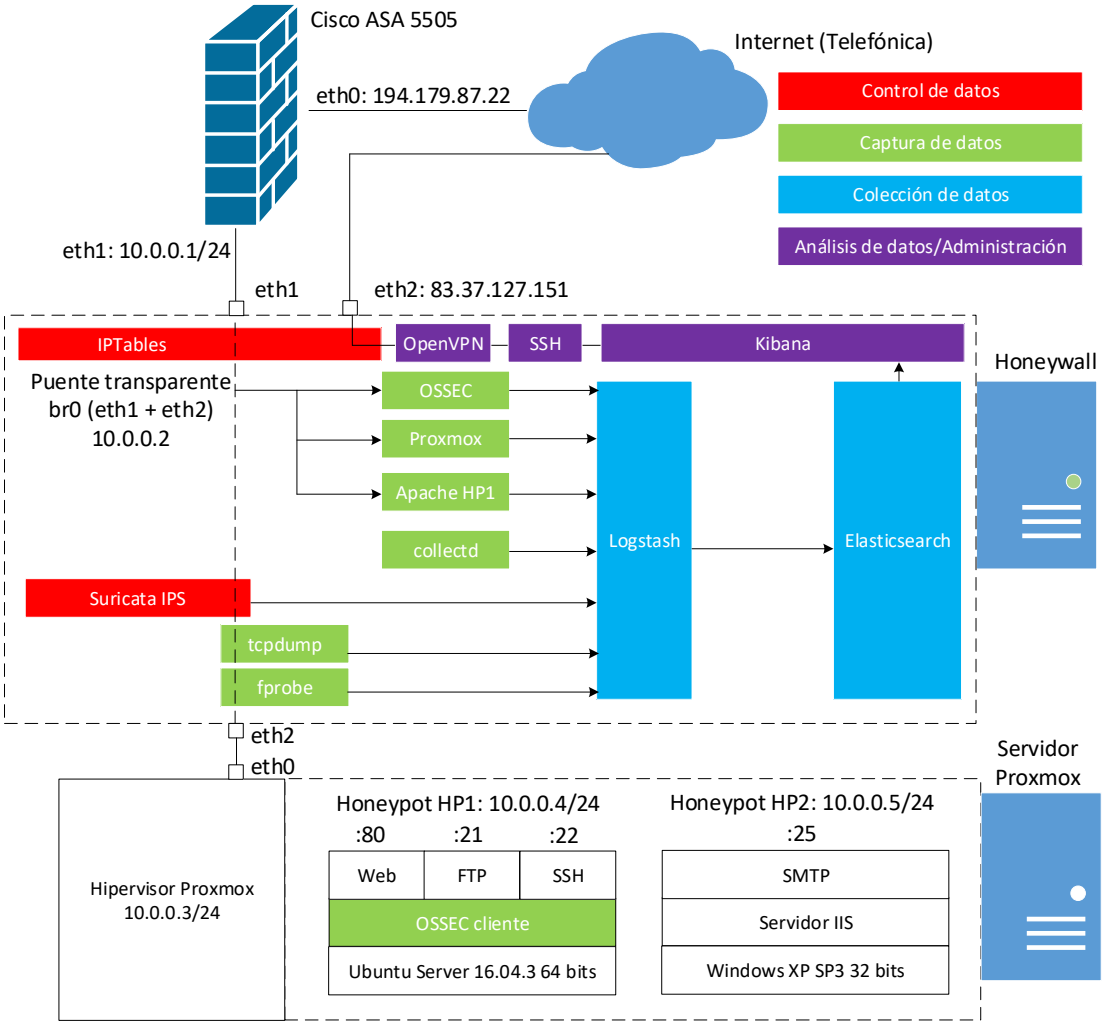

Figura 56 Honeypot HP2 en la honeynet

Así mismo, se configuran en Kibana las gráfica siguientes para datos relacionados con el honeypot HP2 en cuestión:

- Nombre: HP2 - Stat – CPU: histórico del uso de CPU del honeypot HP2

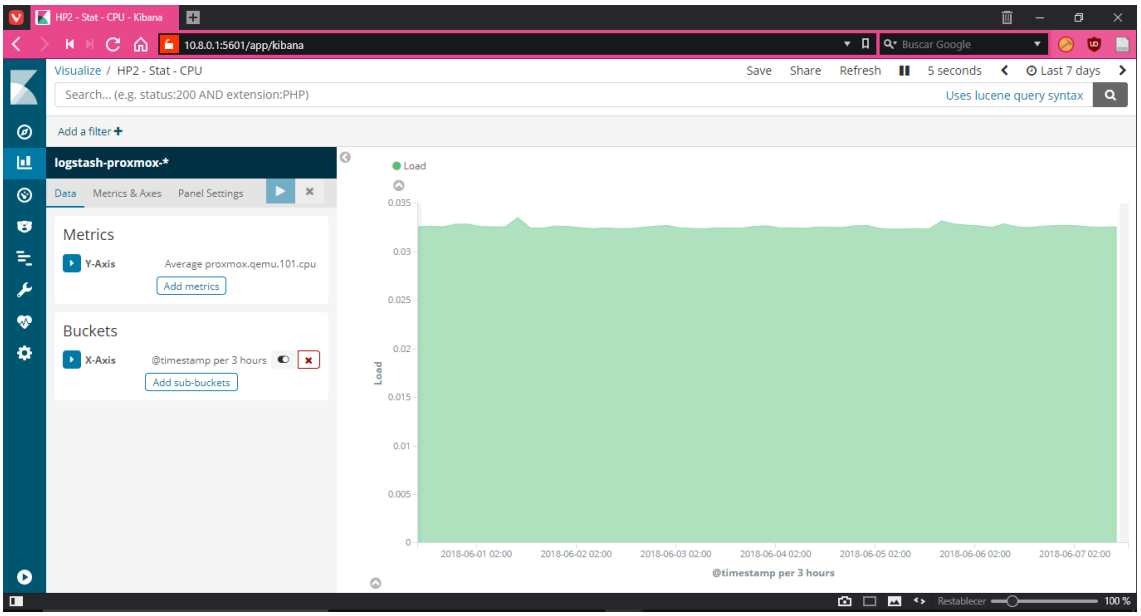

Figura 57 Gráfica HP2 - Stat – CPU

- Nombre: HP2 - Stat – Mem: histórico del uso de memoria RAM del honeypot HP2

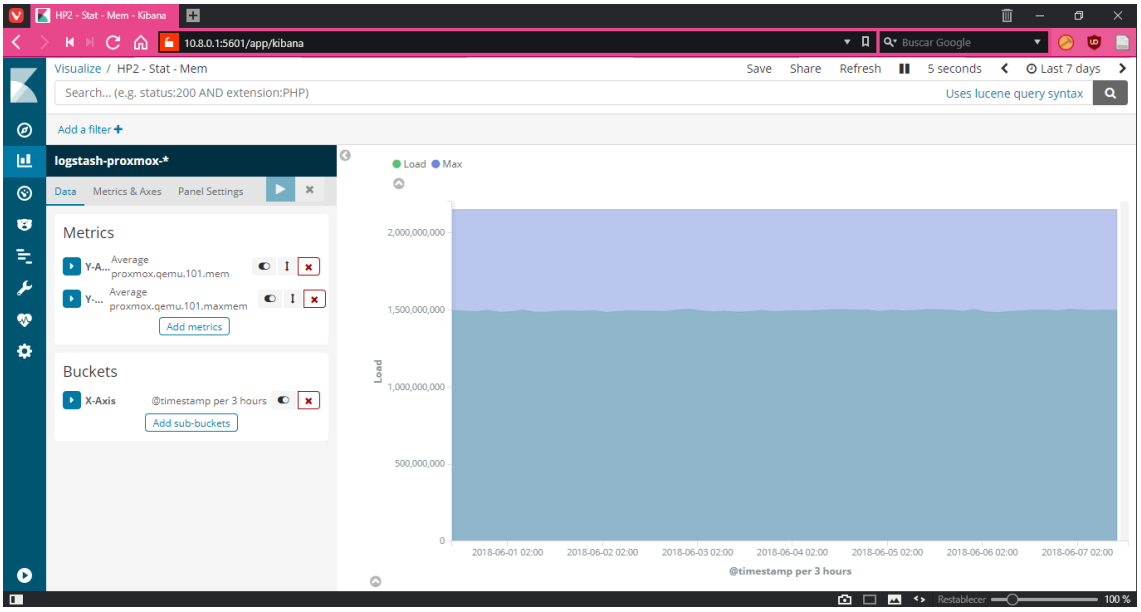

Figura 58 Gráfica HP2 - Stat – Mem

- Nombre: Netflow - Bridge - Stat - HP2: histórico del tráfico del puente de red relacionado con el honeypot HP2

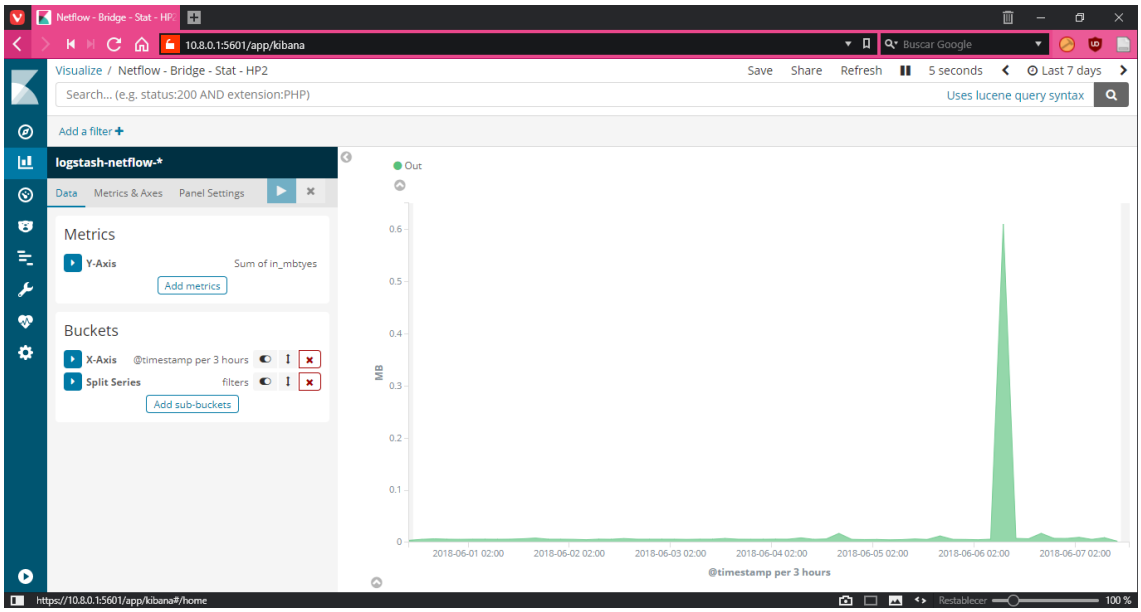

Figura 59 Gráfica Netflow - Bridge - Stat - HP2

- Nombre: Netflow - HP2 – In: gráfico de tarta de relaciones de IPs origen y puertos del honeypot HP2 para conexiones entrantes, ordenadas por Mb transferidos.

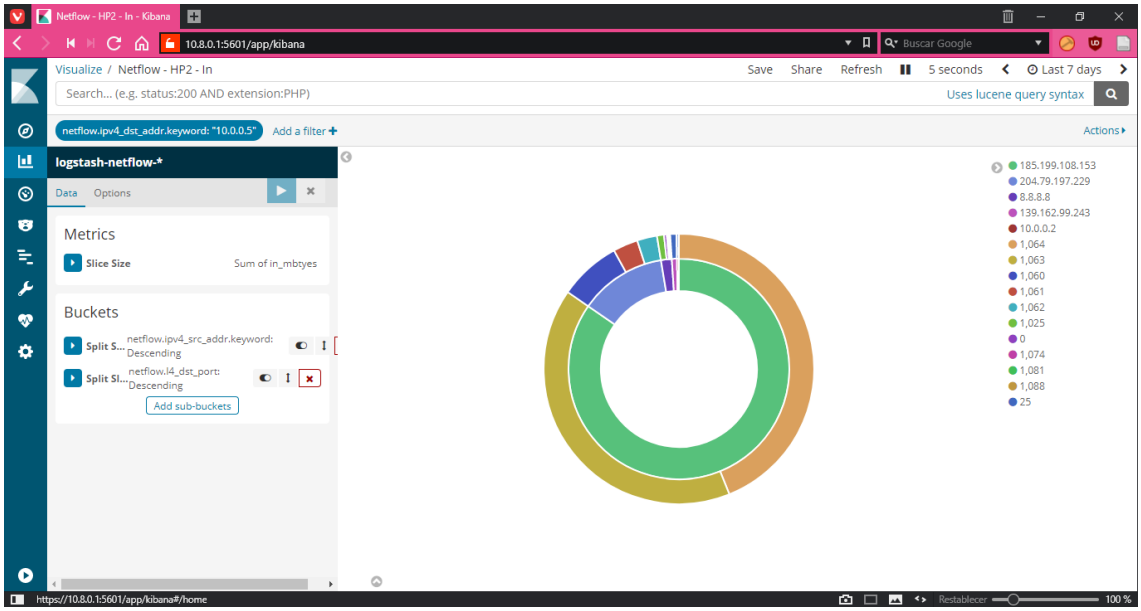

Figura 60 Gráfica Netflow - HP2 – In

- Nombre: Netflow - HP2 – Out: gráfico de tarta de relaciones de IPs destino y puertos del honeypot HP2 para conexiones salientes, ordenadas por Mb transferidos.

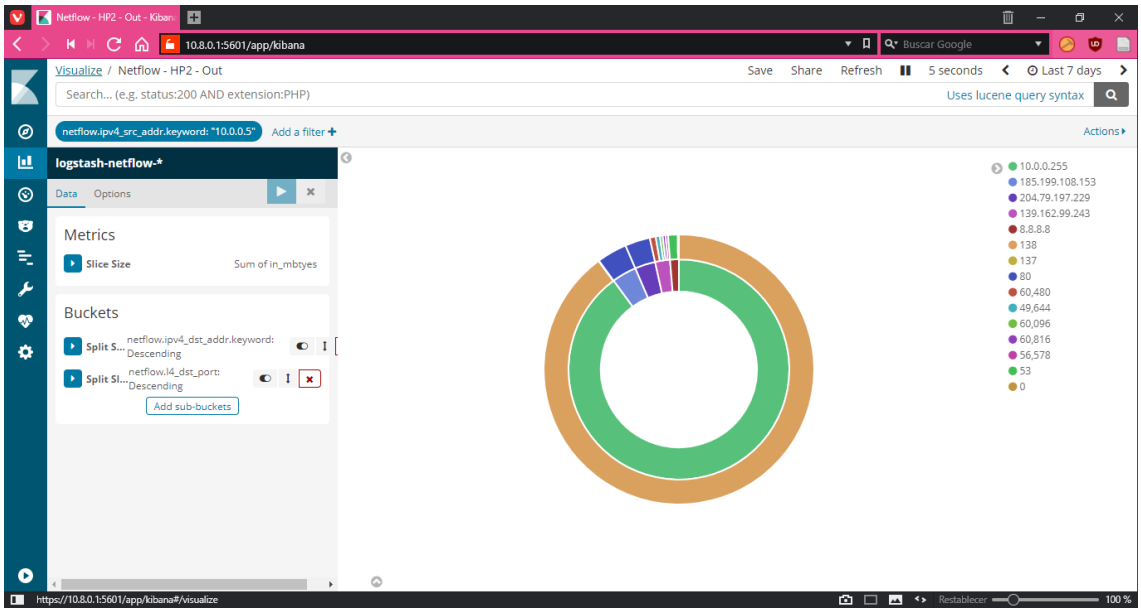

Figura 61 Gráfica Netflow - HP2 – Out

- Nombre: HP2 - SMTP - Count IP: histórico de las del número de correos enviados desde diferentes IPs origen al servidor SMTP del honeypot HP2.

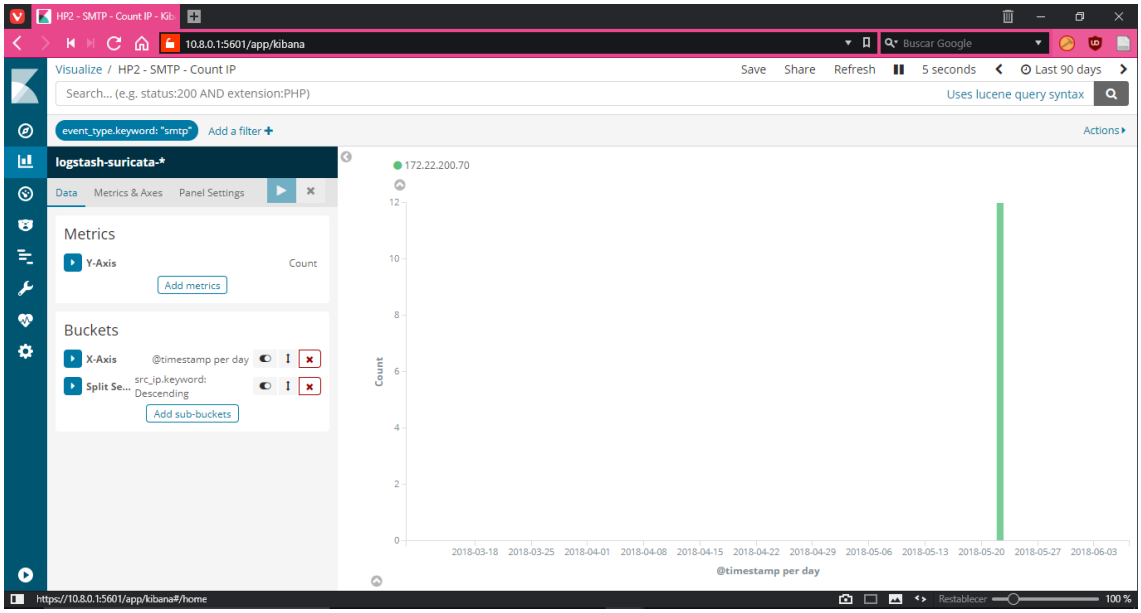

Figura 62 Gráfica HP2 - SMTP - Count IP

- Nombre: HP2 - SMTP - From – IP: gráfico de tarta de relaciones de direcciones de correo remitentes con IPs origen del correo, ordenadas por número de apariciones.

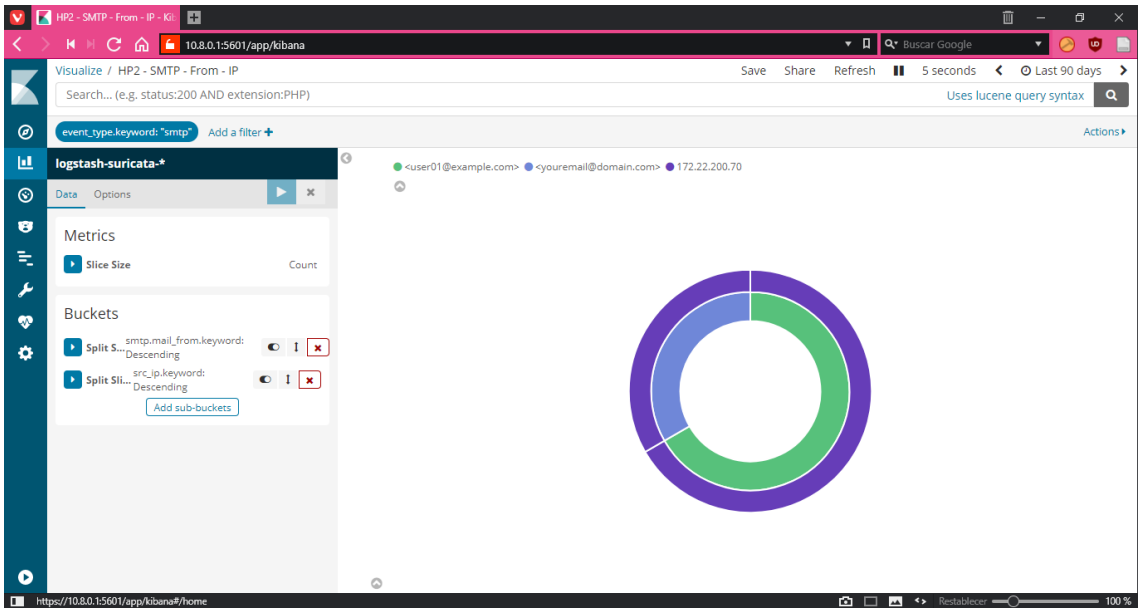

Figura 63 Gráfica HP2 - SMTP - From – IP

- Nombre: HP2 - SMTP - From address: gráfica de los remitentes de correo más repetidos.

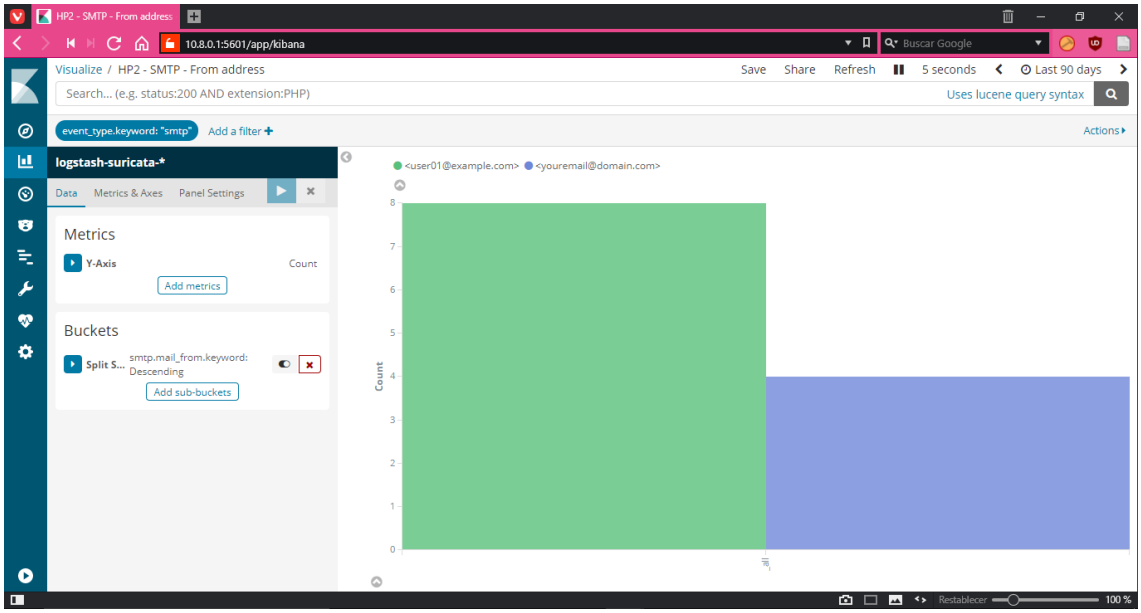

Figura 64 Gráfica HP2 - SMTP - From address

- Nombre: HP2 - SMTP - Rcpt address: gráfica de los destinatarios de correo más repetidos.

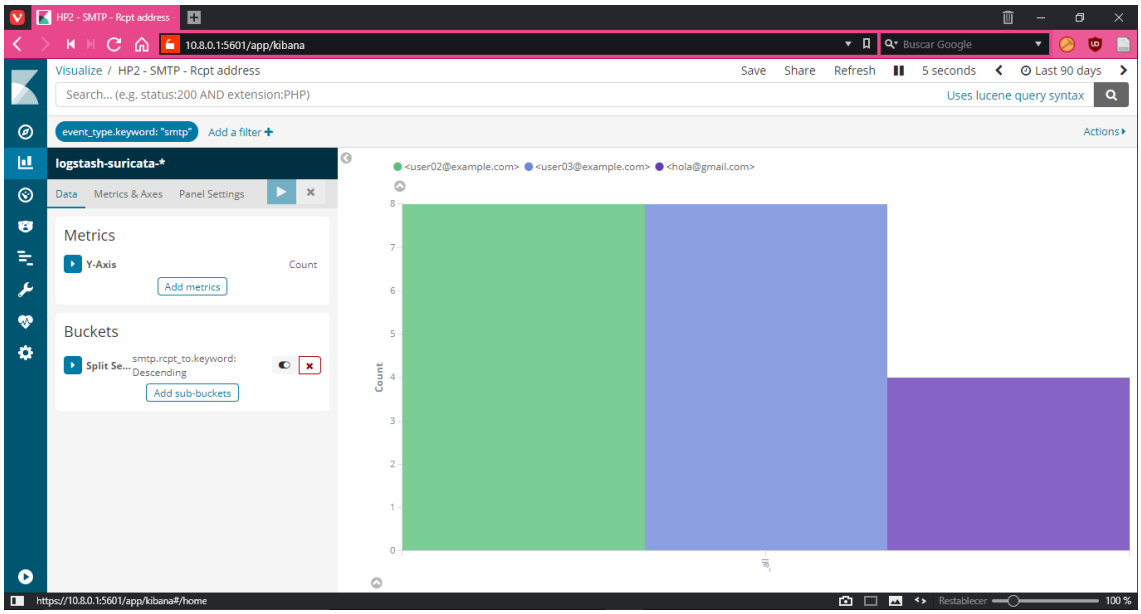

Figura 65 Gráfica HP2 - SMTP - Rcpt address

9.7 Paneles de datos en Kibana

Una vez configuradas todas las gráficas de datos en Kibana, se deben configurar los paneles de visualización. En Kibana, un panel de visualización no es más que un conjunto de gráficas que el usuario desea ver en el panel concreto con la estructura y orden que desee. Basta con, en el apartado Dashboard, añadir un panel nuevo con el nombre que se desee, y añadir las gráficas que se deseen a dicho panel.

En este proyecto, se crean los paneles siguientes (se muestra en imagen sólo el panel principal debido al gran tamaño de los paneles):

- Nombre: Main: panel principal de administración con las siguientes gráficas:
  - Netflow - Bridge – MB
  - NIPS - Event type
  - Netflow - Bridge - Stat - HP1
  - Netflow - Bridge - Stat - HP2
  - HWStat – CPU
  - HP1 - Stat – CPU
  - HP2 - Stat – CPU
  - Proxmox - Stat - CPU

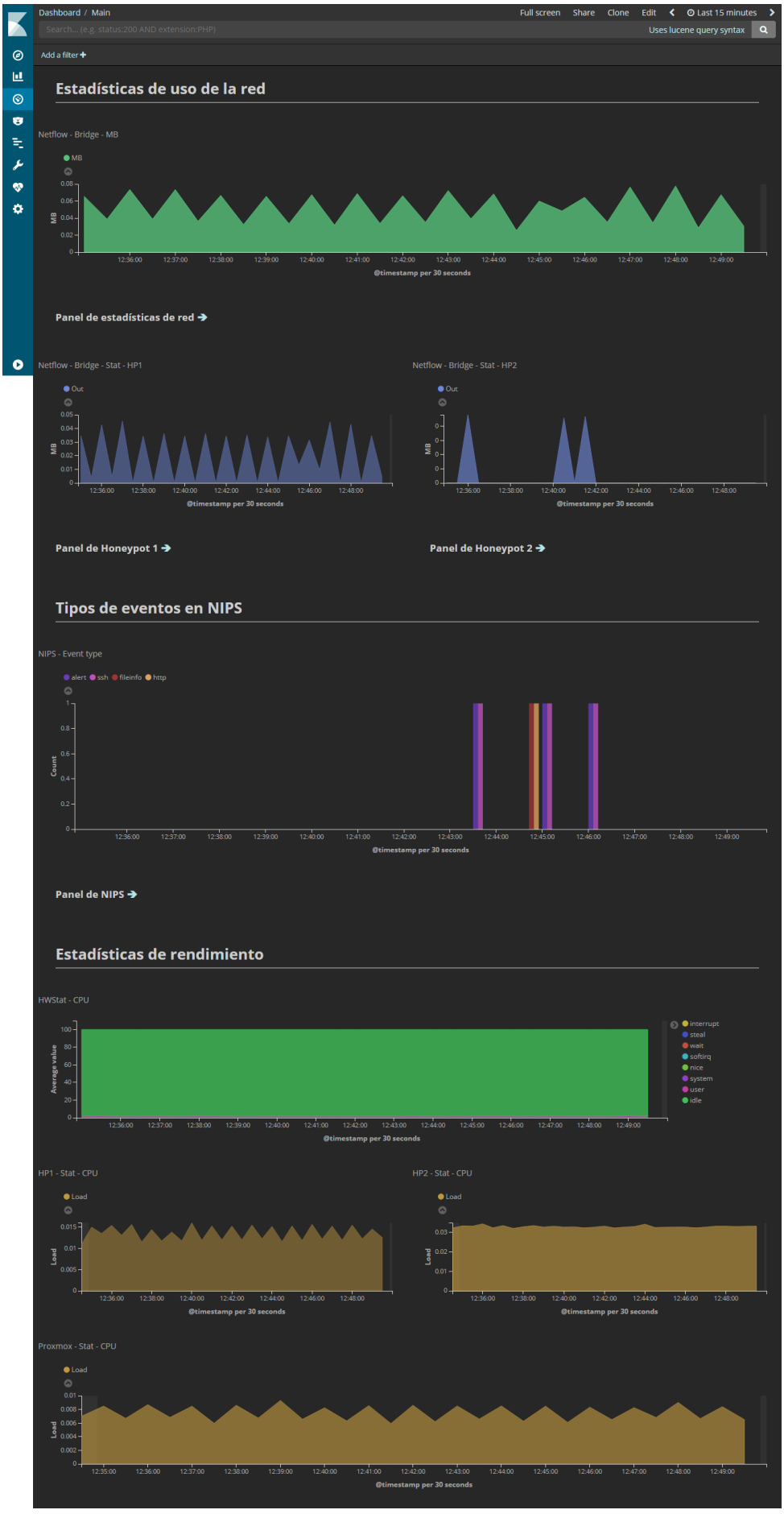
Dashboard / Main
Full screen
Share
Clone
Edit
Last 15 minutes
Search... (e.g. status:200 AND extension:PHP)
Uses lucene query syntax
Add a filter
Estadísticas de uso de la red
Netflow - Bridge - MB
MB
@timestamp per 30 seconds
Panel de estadísticas de red
Netflow - Bridge - Stat - HP1
Out
Netflow - Bridge - Stat - HP2
Panel de Honeypot 1
Panel de Honeypot 2
Tipos de eventos en NIPS
NIPS - Event type
alert
ssh
fileinfo
http
Count
Panel de NIPS
Estadísticas de rendimiento
HWStat - CPU
Average value
interrupt
steal
wait
softirq
nice
system
user
idle
HP1 - Stat - CPU
HP2 - Stat - CPU
Load
Proxmox - Stat - CPU

Figura 66 Panel de visualización principal de Kibana

- Nombre: Netflow: panel de estadísticas de red que contiene las siguientes gráficas:
  - Netflow – Búsqueda
  - Netflow - Bridge – MB
  - Netflow - HP1 – Out
  - Netflow - HP1 – In
  - Netflow - Bridge - Stat - HP1
  - Netflow - HP2 – In
  - Netflow - HP2 – Out
  - Netflow - Sesiones
  - Datos en crudo del índice logstash-netflow-{date}
- Nombre: NIPS: panel de gráficas sobre la prevención de intrusiones en red de Suricata:
  - NIPS – Búsqueda
  - NIPS - Event type
  - NIPS – AvsB
  - NIPS - Top 5 Signatures Allowed
  - NIPS - Top 5 Signatures Blocked
  - NIPS - Top 10 IP-Ports Allowed
  - NIPS - Top 10 IP-Ports Blocked
  - Datos en crudo del índice logstash-suricata-{date}
- Nombre: HP1: panel de gráficas para datos relacionados con el honeypot HP1:
  - HP1 - Stat – CPU
  - HP1 - Stat – Mem
  - HP1 - SSH - Success root and user
  - HP1 - SSH - Top failed users
  - HP1 - SSH - Failed – IP
  - HP1 - FTP - Login OK
  - HP1 - FTP - Login Failed
  - HP1 - FTP - Files DUD
  - HP1 - FTP - Failed – IP
  - HP1 - Apache – UA
  - HP1 - Apache - Top 10 IPs
  - HP1 - Apache – GETvsPOST
  - HP1 - Apache - Top 10 Path
- Nombre: HP1 – SSH: panel de gráficas del servidor SSH del honeypot HP1:
  - HP1 - SSH - Success root and user
  - HP1 - SSH - Top failed users
  - HP1 - SSH - Failed – IP
  - Datos en crudo del servidor SSH
- Nombre: HP1 – FTP: panel de gráficas del servidor FTP del honeypot HP1:
  - HP1 - FTP - Login OK
  - HP1 - FTP - Login Failed
  - HP1 - FTP - Files DUD

  - HP1 - FTP - Failed – IP
  - Datos en crudo del servidor FTP
- Nombre: HP1 – Web: panel de gráficas del servidor web del honeypot HP1:
  - HP1 - Apache – UA
  - HP1 - Apache - Top 10 IPs
  - HP1 - Apache – GETvsPOST
  - HP1 - Apache - Top 10 Path
  - Datos en crudo del servidor Web
- Nombre: HP2: panel de gráficas de datos relacionados con el honeypot HP2:
  - HP2 - Stat – Mem
  - HP2 - Stat – CPU
  - HP2 - SMTP - Count IP
  - HP2 - SMTP - From – IP
  - HP2 - SMTP - From address
  - HP2 - SMTP - Rcpt address
  - Datos en crudo del servidor SMTP

9.8 Resumen de herramientas

A modo de resumen, se muestran la siguiente tablas que incluyen todas las herramientas y datos relacionados con los mecanismos de control, captura y análisis de datos, así como de administración de la honeynet implementados en este proyecto.

9.8.1 Control de datos

Herramienta	Origen	Destino	Descripción
IPTables	Internet	Honeypots	Control y limitación de conexiones
Suricata	Internet	Honeypots	NIPS sobre tráfico de honeypots

Tabla 4 Resumen de control de datos

9.8.2 Captura de datos

Herramienta	Origen	Destino	Descripción
Suricata	Honeywall	Fichero en honeywall /var/log/suricata/eve.json	Datos generados por NIPS en puente de red
OSSEC	Clientes en honeypots	Fichero en honeywall /var/ossec/alerts/alerts.json	HIDS en honeypots
Script	Honeypots	Ficheros en honeypots /var/log/script/script-*.log	Registro de comandos emitidos en honeypots
Tcpdump	Honeywall	Ficheros en honeywall en /var/log/tcpdump/...	Captura de tráfico del puente de red
Fprobe	Honeywall	Honeywall: 127.0.0.1:2055	Datos estadísticos y de sesión de la honeynet
Collectd	Honeywall	Honeywall: 127.0.0.1:25826	Datos estadísticos de uso del honeywall
Proxmox	Servidor Proxmox	Honeywall: 10.0.0.2:2003	Estadísticas de servidor de virtualización y honeypots

Tabla 5 Resumen de captura de datos

9.8.3 Colección de datos

Herramienta	Origen	Medio de transmisión	Destino

Rsync	Honeypots	Conexión unidireccional por SSH a los honeypots	Ficheros en honeywall /var/log/hp{1,2,..,n}
Logstash	Suricata	Fichero en honeywall /var/log/suricata/eve.json	Elasticsearch: 127.0.0.1:9001
	OSSEC	Fichero en honeywall /var/ossec/alerts/alerts.json	Elasticsearch: 127.0.0.1:9001
	Tcpdump	Honeywall: 127.0.0.1:2055	Elasticsearch: 127.0.0.1:9001
	Collectd	Honeywall: 127.0.0.1:25826	Elasticsearch: 127.0.0.1:9001
	Servidor Proxmox	Honeywall: 10.0.0.2:2003	Elasticsearch: 127.0.0.1:9001
	Servidor Web	Fichero en honeywall: /var/log/hp1/apache2/access.log	Elasticsearch: 127.0.0.1:9001

Tabla 6 Resumen de colección de datos

9.8.4 Análisis de datos y administración

Herramienta	Origen	Destino	Datos	Descripción
Servidor VPN	Internet	Honeywall	No aplica	Acceso administrativo por túnel cifrado
Servidor SSH	Internet (a través de VPN)	Honeywall	Ficheros logs en crudo	Acceso por consola al honeywall
Kibana	Internet (a través de VPN)	Honeywall	Datos almacenados en Elasticsearch	Administración de Elasticsearch y Logstash
Hipervisor Proxmox	Internet (a través de VPN)	Proxmox (a través de NAT en honeywall)	Datos estadísticos de las máquinas virtuales	Administración de y acceso a consolas de máquinas virtuales

Tabla 7 Resumen de análisis de datos y administración

9.9 Resumen de comunicaciones

A modo de resumen, en este apartado se especifican de manera simple todos las vías de comunicación con la honeynet, tanto para los honeypots, como para la administración de los elementos que la componen.

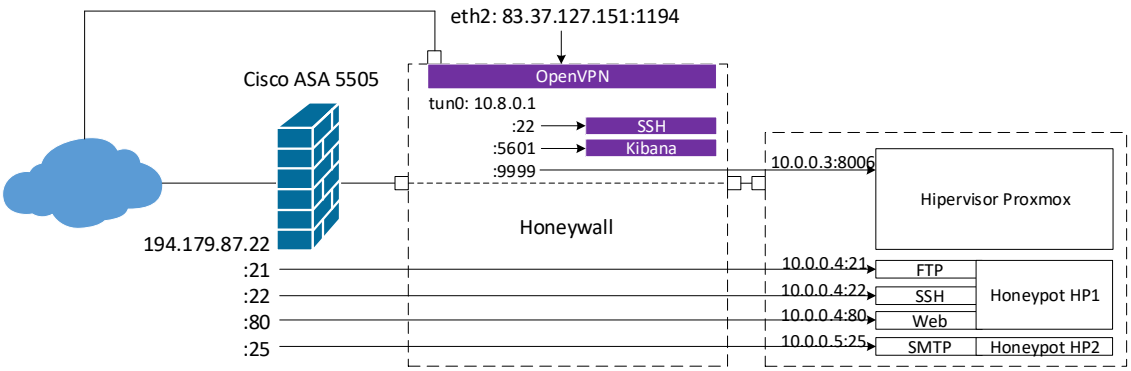

Figura 67 Resumen de comunicaciones

El acceso público a la honeynet para el ataque a los honeypots se realiza a través de la dirección IP pública 194.179.87.22 con la siguiente redirección de puertos

- NAT de la dirección IP 194.179.87.22
    - FTP por puerto 21, redirigido a servidor FTP en 10.0.0.4:21, honeypot HP1.
    - SSH por puerto 22, redirigido a servidor SSH en 10.0.0.4:22, honeypot HP2.
    - HTTP por puerto 80, redirigido a servidor HTTP en 10.0.0.4:80, honeypot HP1.
    - SMTP por puerto 25, redirigido a servidor SMTP en 10.0.0.5:25, honeypot HP2.

Mediante la página web de Pentest-Tools [13], disponemos de la posibilidad de hacer un escaneo de puertos a la dirección pública que se desee, entre otras acciones. En este caso, se realiza un escaneo de puertos a la IP pública de la honeynet para el acceso a honeypots: 194.179.87.22.

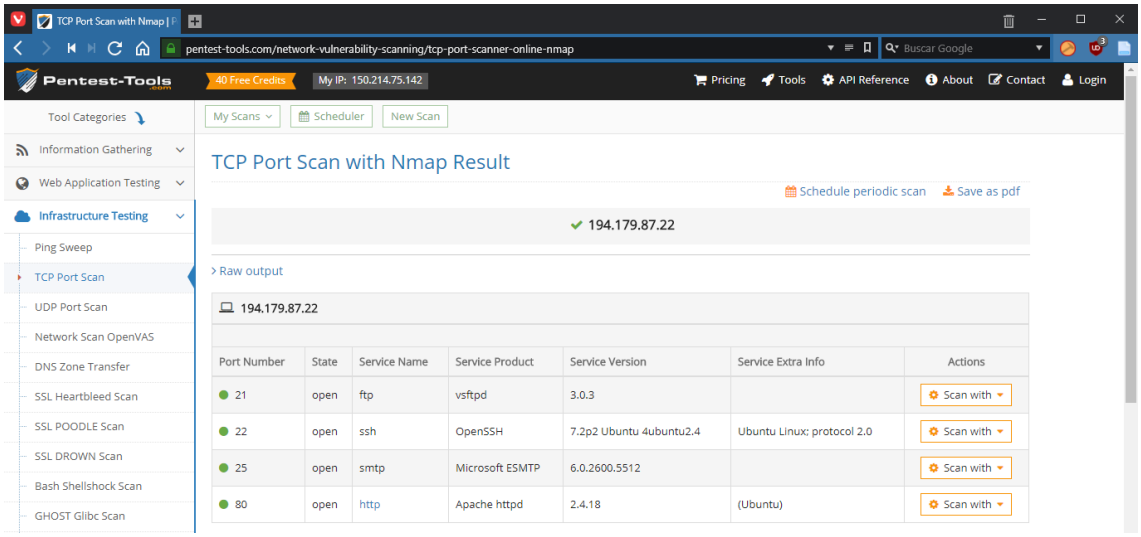

Figura 68 Escaneo de puertos de la honeynet

Se puede ver como se encuentran abiertos los puertos que se han redireccionado hacia los honeypots con los servicios y versiones ya especificados anteriormente respondiendo a dichas conexiones.

Para el acceso administrativo, se debe acceder con el cliente OpenVPN al servidor VPN de la IP pública 83.37.127.151, escuchando en el puerto 1994. Una vez dentro de la VPN, se asigna una dirección dentro de la red privada 10.8.0.0/24 y, a través de la dirección del honeywall perteneciente a dicha red, 10.8.0.1/24, se accede a:

- Servidor SSH del honeywall: 10.8.0.1:22
- Página web de Kibana: https://10.8.0.1:5601
- Página web de Proxmox: https://10.8.0.1:9999

9.10 Resultados

En un intervalo de 15 días que comprenden entre el lunes 21 de mayo de 2018 y el lunes 4 de junio de 2018, se obtienen los siguientes resultados en la honeynet.

9.10.1 Estadísticas del servidor SSH del honeypot HP1

Algunas estadísticas de intentos de inicio de sesión en el servidor SSH del honeypot HP1 son las siguientes:

- Intentos de inicio de sesión fallidos, ordenados por el número de intentos por usuario.

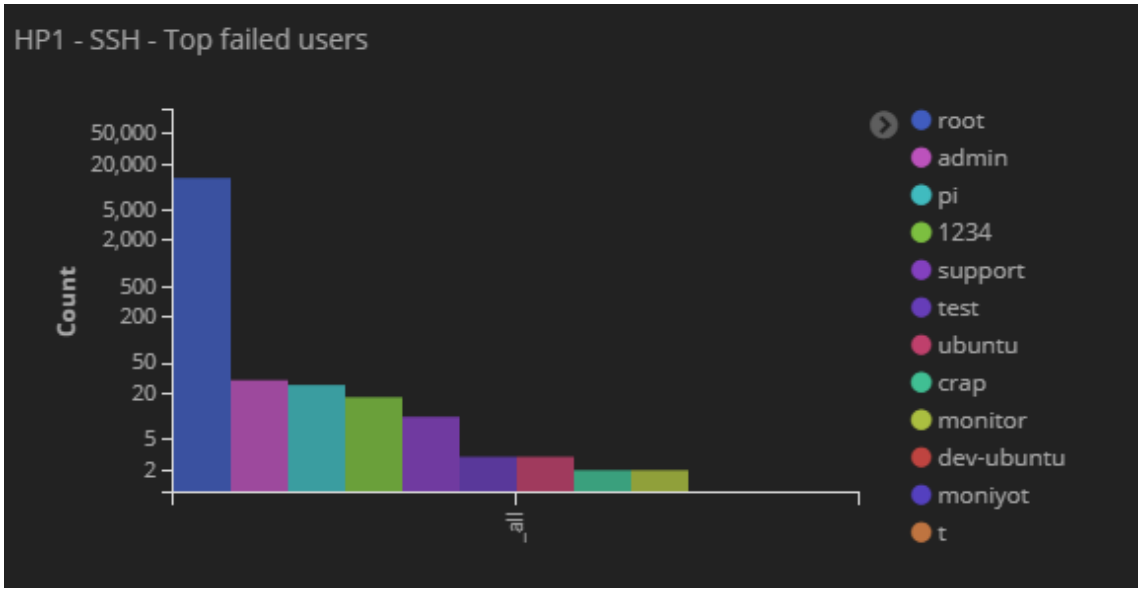

Figura 69 Intentos de inicio de sesión fallidos por SSH

Usuario	Número de intentos
root	13,146
admin	30
pi	26
1234	18
support	10
test	3
ubuntu	3
crap	2
monitor	2
dev-ubuntu	1
moniyot	1
t	1

Tabla 8 Intentos de inicio de sesión fallidos por SSH

- Intentos de inicio de sesión fallidos, ordenados por el número de intentos por usuario y dirección IP origen del intento.

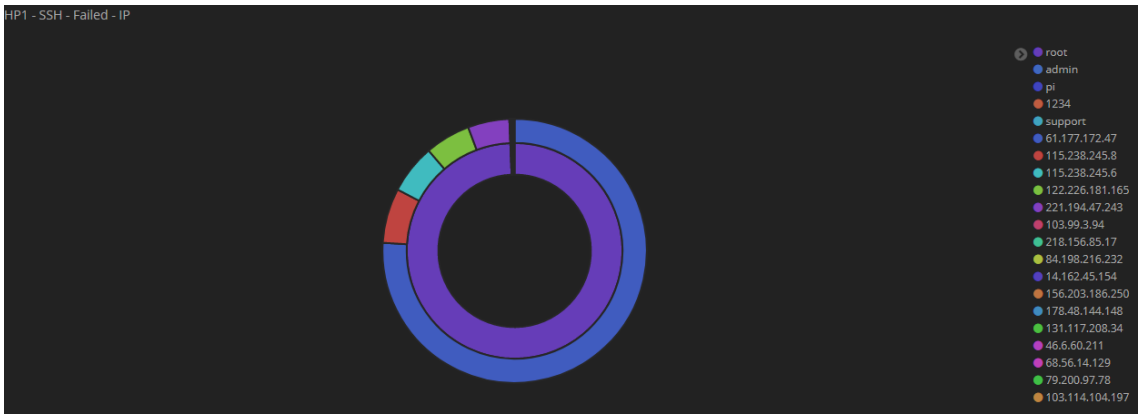

Figura 70 Intentos de inicio de sesión fallidos por SSH con direcciones IP

Dirección IP	Usuario	Número de intentos
61.177.172.47	root	6,727
115.238.245.8	root	594
115.238.245.6	root	535
122.226.181.165	root	492
221.194.47.243	root	450
103.99.3.94	1234	10
	admin	10
	support	10
218.156.85.17	admin	7
84.198.216.232	admin	7
14.162.45.154	admin	2

Tabla 9 Intentos de inicio de sesión fallidos por SSH con direcciones IP

De estas estadísticas se puede observar como la mayoría de los ataques de intento de inicio de sesión por SSH suelen ser llevados a cabo mediante herramientas de fuerza bruta, debido a que el número de intentos de inicio de sesión como usuario *root* desde la dirección IP 61.177.172.47 no corresponden a un ataque manual.

9.10.2 Estadísticas del servidor FTP del honeypot HP1

Algunas estadísticas de intentos de inicio de sesión en el servidor FTP del honeypot HP1 son las siguientes:

- Intentos de inicio de sesión fallidos, ordenados por el número de intentos por usuario.

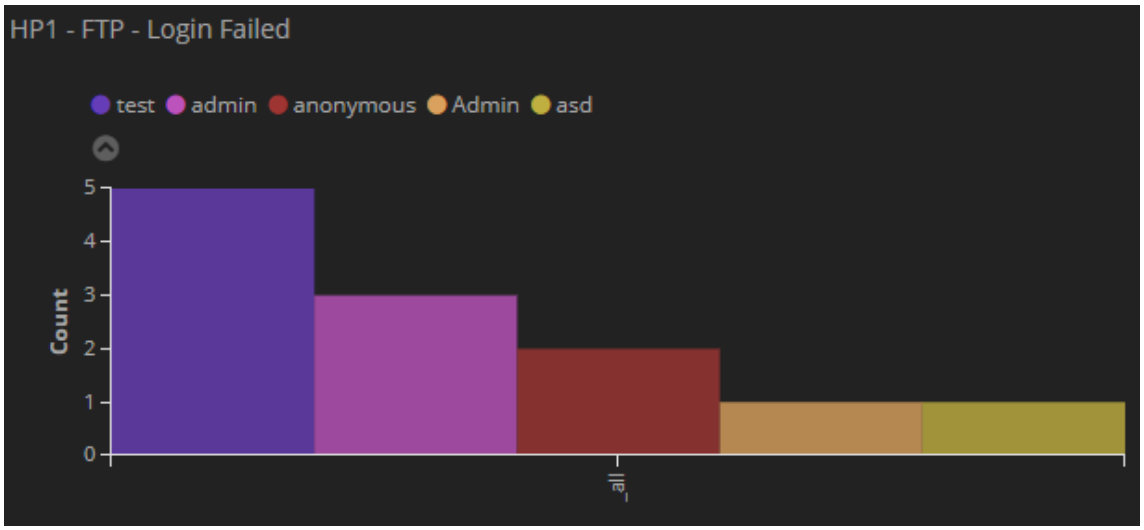

Figura 71 Intentos de inicio de sesión fallidos por FTP

Usuario	Número de intentos
test	5
admin	3
anonymous	2
Admin	1
asd	1

Tabla 10 Intentos de inicio de sesión fallidos por FTP

- Intentos de inicio de sesión fallidos, ordenados por el número de intentos por usuario y dirección IP origen del intento.

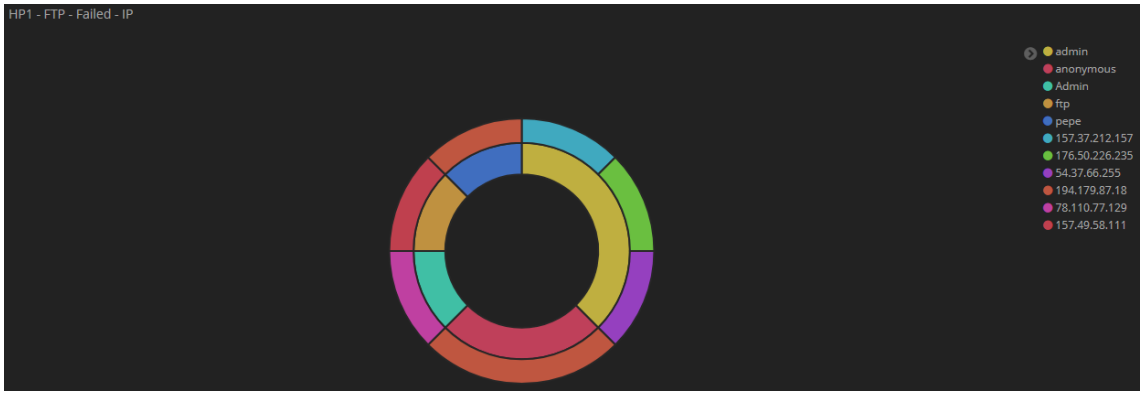

Figura 72 Intentos de inicio de sesión fallidos por FTP con direcciones IP

Dirección IP	Usuario	Número de intentos
157.37.212.157	admin	3
176.50.226.235	admin	3
54.37.66.255	admin	3
194.179.87.18	anonymous	2
	pepe	1
78.110.77.129	Admin	1
157.49.58.11	ftp	1

Tabla 11 Intentos de inicio de sesión fallidos por FTP con direcciones IP

En comparación con las ataques al servicio SSH del honeypot HP1, el servidor FTP recibe menos ataques, normalmente debido a que un inicio de sesión por SSH devuelve al atacante una sesión totalmente interactiva, mientras que una sesión FTP está un poco más restringida. Aun así, el servicio FTP tampoco está libre de recibir ataques de fuerza bruta para intentar adivinar el usuario/contraseña de inicio de sesión

9.10.3 Estadísticas del servidor web en el honeypot HP1

Algunas estadísticas relacionadas con el servidor web del honeypot HP1 son las siguientes:

- Direcciones IPs que más peticiones GET/POST realizan al servidor web.

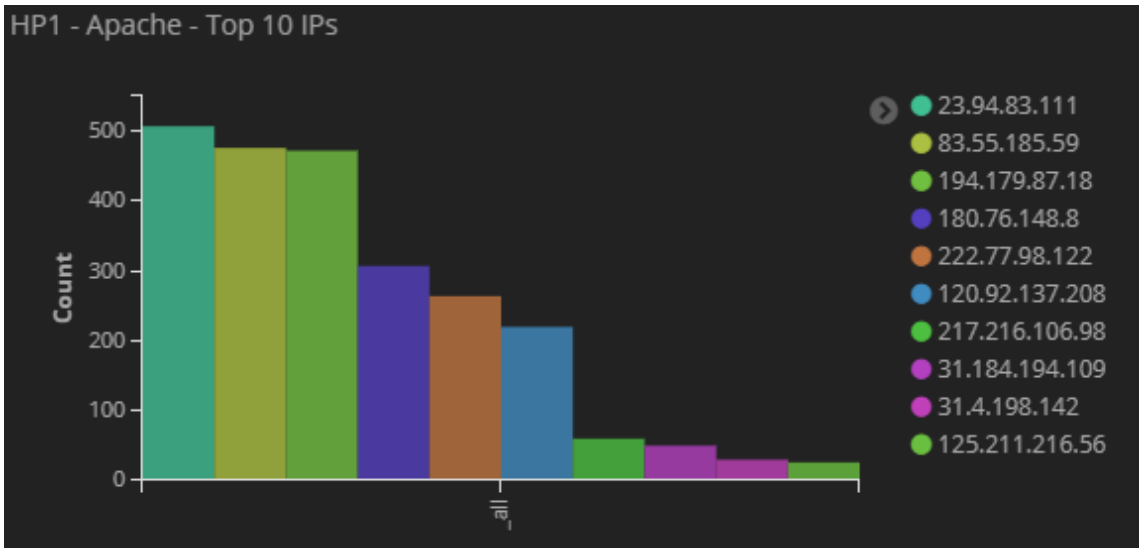

Figura 73 Direcciones IPs que más peticiones GET/POST realizan al servidor web

Dirección IP	Número de peticiones
23.94.83.111	507
83.55.185.59	476
194.179.87.28	472
180.76.148.8	307
222.77.98.122	264
120.92.137.208	220
217.216.106.98	60
31.184.194.109	50
31.4.198.142	30
125.211.216.56	26

Tabla 12 Direcciones IPs que más peticiones GET/POST realizan al servidor web

- URLs más solicitadas del servidor web

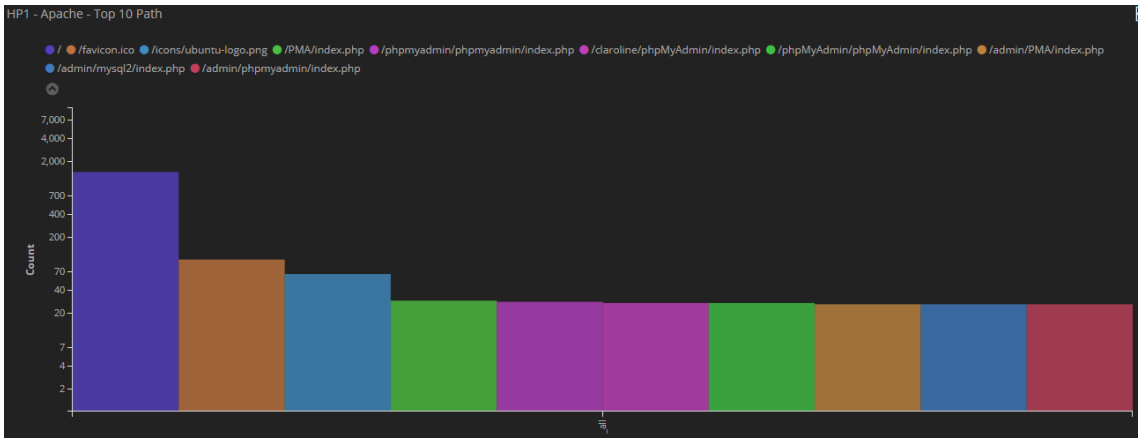

Figura 74 URLs más solicitadas del servidor web

URL	Número de peticiones
/	1,442
/favicon.ico	101
/icons/ubuntu-logo.png	65
/PMA/index.php	29
/phpmyadmin/phpmyadmin/index.php	28
/claroline/phpMyAdmin/index.php	27
/phpMyAdmin/phpMyAdmin/index.php	26
/admin/PMA/index.php	26
/admin/mysql2/index.php	26
/admin/phpmyadmin/index.php	26

Tabla 13 URLs más solicitadas del servidor web

En estas estadísticas se puede observar que el servidor web es un vector de ataque muy común en los ataques automáticos que se llevan a cabo hoy en día y, de las estadísticas de URLs más solicitadas se puede observar como los atacantes buscan las localizaciones por defecto de los paneles de administración de herramientas web que se pueden encontrar en entornos de producción en Internet hoy en día, siendo la mayoría, intentos de entrar en el servicio PHPMyAdmin, de administración de bases de datos MySQL a través de una interfaz web.

9.10.4 Estadísticas de Suricata

Las estadísticas de alertas lanzadas por Suricata, con tráfico permitido y bloqueado, son las siguientes

- Comparación del número de alertas con tráfico permitido con el número de alertas con tráfico bloqueado.

Figura 75 Tráfico permitido vs tráfico bloqueado por Suricata

- Alertas de Suricata más repetidas con tráfico permitido

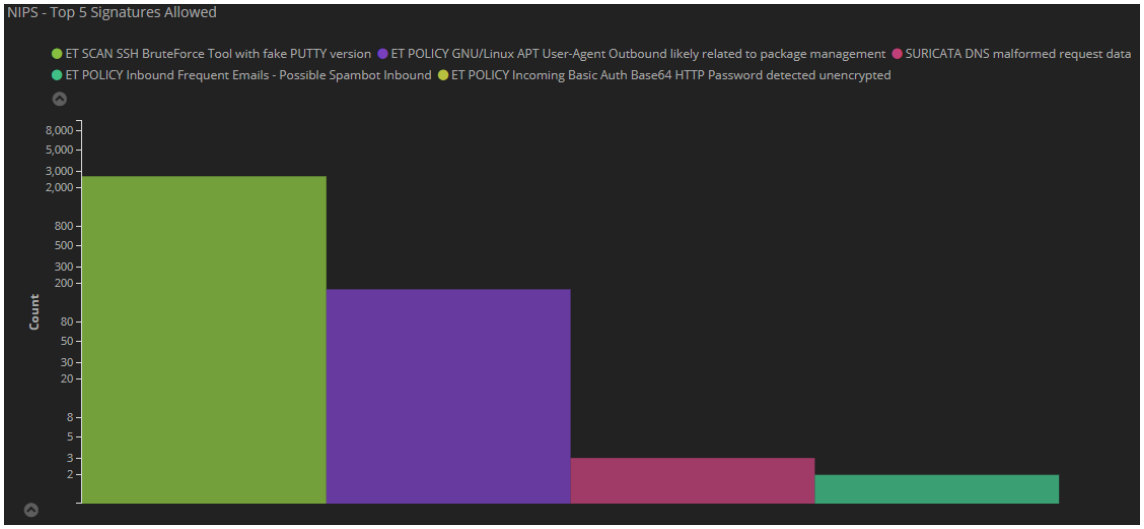

Figura 76 Alertas de Suricata más repetidas con tráfico permitido

Regla	Alertas
ET SCAN SSH BruteForce Tool with fake PUTTY version	2,644
ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management	174
SURICATA DNS malformed request data	3
ET POLICY Inbound Frequent Emails - Possible Spambot Inbound	2
ET POLICY Incoming Basic Auth Base64 HTTP Password detected unencrypted	1

Tabla 14 Alertas de Suricata más repetidas con tráfico permitido

- Alertas de Suricata más repetidas con tráfico bloqueado.

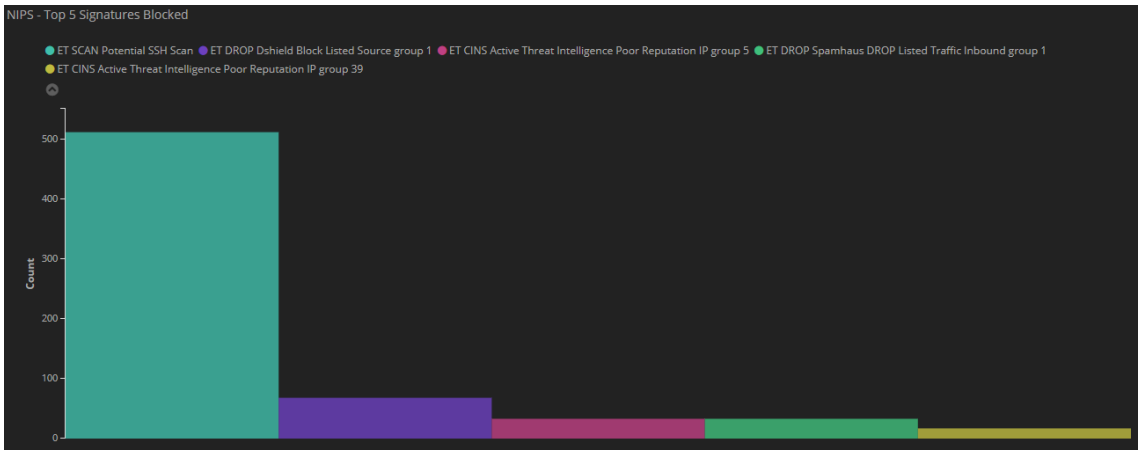

Figura 77 Alertas de Suricata más repetidas con tráfico bloqueado

Regla	Alertas
ET SCAN Potential SSH Scan	511
ET DROP Dshield Block Listed Source group 1	68
ET CINS Active Threat Intelligence Poor Reputation IP group 5	33
ET DROP Spamhaus DROP Listed Traffic Inbound group 1	33
ET CINS Active Threat Intelligence Poor Reputation IP group 39	17

Tabla 15 Alertas de Suricata más repetidas con tráfico bloqueado

De estas estadísticas se puede ver que, aunque Suricata bloquee tráfico de IPs que se conocen como maliciosas (grupos de mala reputación), hay muchas direcciones IPs permitidas que realizan ataques automáticos y no automáticos que no se reportan como maliciosas, por lo que una buena primera aportación del proyecto sería compartir las IPs detectadas como atacantes comunidades global abiertas de compartición de amenazas.

Cabe destacar que en el periodo de tiempo establecido, no se han detectado ataques del día cero exitosos en la honeynet.

10 Planificación temporal

La planificación temporal del proyecto de diseño de una honeynet virtual para la investigación de ataques informáticos en la red de la Diputación de Cádiz es las siguiente:

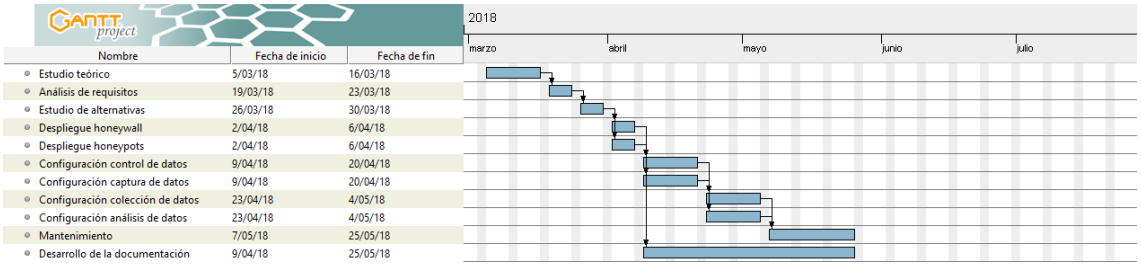

Figura 78 Planificación temporal del proyecto

11 Resumen del Presupuesto

El resumen del presupuesto para el proyecto desarrollado se encuentra especificado en la siguiente tabla:

	Precio (€)
Cableado	8,64
Conexionado a Internet	107.70
Hardware	4.395,08
Autor del proyecto	2.105,25
Total:	**6.616,67**

Tabla 16 Resumen del presupuesto

12 Orden de prioridad de los documentos

El orden de prioridad de lectura de los documentos para este proyecto es el siguiente:

1. Especificaciones.
2. Mediciones.
3. Presupuesto.
4. Memoria.
5. Anexos.
6. Estudio teórico.

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

REF: 0000002

ESTUDIO TEÓRICO

CLIENTE: EMPRESA PROVINCIAL DE INFORMACIÓN DE CÁDIZ S.A. (EPICSA)
PLAZA MADRID S/N, EDIFICIO CARRANZA,
FONDO SUR, LOCAL 10, 11010 CÁDIZ
956261500

AUTOR: CARLOS CARRETERO AGUILAR
INGENIERO INFORMÁTICO
25603515-F
CARLOS.CARRETEROAGUILAR@ALUM.UCA.ES

FIRMADO:
SOLICITANTE AUTOR

CÁDIZ, A 1 DE JULIO DE 2018

1 Ataques informáticos

Un ataque informático es un intento, por parte de un individuo determinado, de tomar el control de un sistema informático, para robar datos privados o para usarlo para su propio beneficio. Un ataque informático también puede tener como objetivo impedir el desempeño habitual de los sistemas informáticos y/o desacreditar a alguna persona determinada u organización.

1.1 Tipos de atacantes

Según las intenciones de los atacantes cuando llevan a cabo un ataque informático, se les puede clasificar en tres grupos: sombrero blanco, sombrero gris y sombrero negro.

Tipo de atacante	Sombrero blanco	Sombrero negro	Sombrero gris
Intención	Buena	Mala	Intermedia
Medios	Legales	Ilegales	A veces ilegales
Objetivo	Auditorías de seguridad	Económico Espionaje industrial Popularidad Diversión	Depende de la intención del ataque

Tabla 17 Tipología de atacantes

1.1.1 Atacantes de sombrero blanco

En inglés *whitehat*, un atacante de sombrero blanco es un atacante con permiso para realizar el ataque a determinados sistemas informáticos. El permiso para realizar los ataques es proporcionado por el propietario/administrador de los sistemas, ya sea una persona o una organización. A estos tipos de ataques se los engloba dentro del denominado Hacking ético.

El objetivo de estos ataques nunca es dañar a una organización, sino realizar una auditoría para detectar fallos de seguridad en los sistemas informáticos y procesos de seguridad implantados, con el objetivo de su reparación para evitar una posible explotación malintencionada en el futuro.

1.1.2 Atacantes de sombrero negro

En inglés *blackhat*, un atacante de sombrero negro es un atacante con malas intenciones que ataca sistemas informáticos sin permiso del propietario de este por medios ilegales. El objetivo del ataque puede variar según la motivación del atacante:

- Económico, si el atacante puede obtener beneficio de la venta de información.
- Espionaje industrial, si el atacante es contratado por la competencia de la organización objetivo.
- Popularidad, si el atacante puede publicitar su autoría del ataque para reconocimiento de la comunidad de atacantes.
- Diversión.

Los atacantes de sombrero negro pueden actuar en solitario o pertenecer a organizaciones estructuradas que funcionan de manera similar a otras organizaciones criminales, como, por ejemplo, mafias. Dentro de los atacantes malintencionados también podemos encontrarnos a uno de los tipos de atacantes comunes, los internos. Un atacante interno es un empleado

actual de la empresa, o un exempleado con acceso a los sistemas, cuyo objetivo es dañar a la empresa, ya sea por desacuerdo en algún aspecto del trabajo, falta de actitud positiva o despido inminente. Los ataques llevados a cabo por este tipo de atacantes son los que menos se prevén y los que más daño pueden provocar a una organización.

1.1.3 Atacantes de sombrero gris

En inglés *greyhat*, un atacante de sombrero gris es un atacante que se sitúa en medio de los dos tipos de atacantes anteriores. Un atacante de sombrero gris no tiene malas intenciones, pero realiza ataques informáticos sin permiso explícito del propietario/administrador de los sistemas informáticos objetivo, usualmente por medios ilegales.

El objetivo final del atacante de sombrero gris puede ser informar a la persona u organización propietario del sistema informático atacado para alertar de fallos de seguridad. Al mismo tiempo, el atacante podría alertar de dichos fallos a la comunidad de atacantes de sombrero negro y observar los resultados por pura diversión.

Los atacantes de sombrero gris también pueden contener un subgrupo de atacantes a los que se denomina *Script-kiddies*. Este grupo de atacantes se caracteriza por la falta de conocimiento en seguridad de sus componentes y de la falta de motivación en sus actividades. Se podría decir que son atacantes que descubren herramientas de seguridad y lanzan ataques de seguridad sin ningún tipo de conocimiento de su forma de actuar ni de las consecuencias que podría tener. Aun así, este tipo de atacantes pueden causar graves daños, aunque no sea su intención.

1.2 Tipos de ataques informáticos

Los ataques informáticos se pueden clasificar en varios grupos, según su objetivo.

Tipo de ataque	Reconocimiento	Acceso	Denegación de servicio
Técnicas	Internet	Descubrimiento de contraseñas	Acceso previo
	Pings	Puertas traseras	Saturación
	Escaneo de puertos	Exploits	Malformación de tráfico
	Escaneo de vulnerabilidades	Escalada de privilegios	

Tabla 18 Tipos de ataques informáticos

1.2.1 Reconocimiento

Los ataques informáticos que tienen el objetivo reconocer los sistemas informáticos objetivo son ataques, que en un principio no dañan los sistemas, pero tienen el objetivo de recabar la máxima información posible sobre el sistema de información objetivo para, así, poder adaptar y optimizar al máximo los ataques futuros.

Dentro de los ataques de reconocimiento podemos encontrar diferentes técnicas:

- **Uso de la información en internet**. Con el uso de herramientas como Google, bases de datos de dominios, registros DNS, etc.

- **Realización de pings**. Mediante el uso sucesivos de la utilidad de red *ping*, un atacante puede descubrir los equipos activos dentro de una subred determinada.
- **Escaneo de puertos**. Mediante esta técnica, un atacante puede descubrir que puertos tiene abiertos un equipo determinado para descubrir que servicios ofrece y descubrir puertos activos determinados que pueden suponer un fallo de seguridad o una vulnerabilidad determinada. Según el tiempo de respuesta en las solicitudes a determinados puertos, se puede identificar el sistema operativo que utiliza el equipo objetivo. La herramienta más conocida para el escaneo de puertos es Nmap.
- **Escaneo de vulnerabilidades**. Mediante esta técnica, un atacante es capaz de descubrir que vulnerabilidades tiene un sistema para así poder explotarlas en ataques sucesivos. Lo más usual es que una herramienta de escaneo de vulnerabilidades utilice bases de datos de vulnerabilidades conocidas como CVE. Existen muchas herramientas de escaneo de vulnerabilidades, siendo OpenVAS una de las más conocidas.

1.2.2 Acceso

Los ataques de acceso son ataques cuyo objetivo final es acceder a un sistema informático con el objetivo de controlar su funcionamiento, robar datos o usarlo como punto de partida de otro ataque informático posterior en contra de otro sistema informático.

Los métodos de acceso que utilizan los ataques pueden variar:

- **Descubrimiento de contraseñas**, mediante técnicas como la ingeniería social, los ataques de diccionario, los ataques de fuerza bruta o la captura de tráfico de red.
- **Puertas traseras**, mediante la instalación de estas por el atacante para accesos posteriores o mediante el uso ilegítimo de una puerta trasera instalada por el fabricante o administrador del sistema objetivo.
- **Explotación de vulnerabilidades**, dependiendo de la tipología de vulnerabilidad, se pueden dar diferentes tipos de ataques. Existen vulnerabilidades de configuración, de calidad de código, de manejo de variables de entorno, de manejo de errores, de validación de entrada, de administración de contraseñas, etc.
- **Escalada de privilegios**, para conseguir un acceso de máximo privilegio a los recursos de un sistema informático.

1.2.3 Denegación de servicio

Los ataques de denegación de servicio (DoS) tienen como objetivo impedir el normal funcionamiento de un sistema informático para le sea imposible continuar con la prestación de servicios. Existen diferentes maneras de lograr denegaciones de servicio.

- **Acceso previo**. Un atacante con acceso al sistema suspende un servicio determinado o apaga el equipo. El atacante también podría ejecutar una gran cantidad de procesos en el sistema para provocar su saturación.
- **Saturación**. Mediante el envío masivo de paquetes de red a un servidor, un atacante puede saturar el proceso de gestión de conexiones del servidor e impedir nuevas conexiones legítimas.

- **Malformación de tráfico**. Un atacante puede aprovechar un fallo de un sistema en la gestión de paquetes de red, para provocar un fallo de procesamiento mediante el envío de un paquete de red malformado.

Los ataques de denegación de servicio también pueden ser lanzados desde un conjunto amplio de máquinas (botnet), con el objetivo de aumentar la eficacia del ataque. A este tipo de denegación de servicio, se le denomina denegación de servicio distribuida (DDoS).

1.3 Ciclo de vida de un ataque informático

Cuando los ataques informáticos son llevados a cabo, estos siguen normalmente un ciclo de vida de diferentes fases en las que cada una de ellas cumple un propósito determinado. En la enciclopedia virtual del fabricante de cortafuegos PaloAlto [14], se definen las siguientes fases del ciclo de vida de un ataque informático:

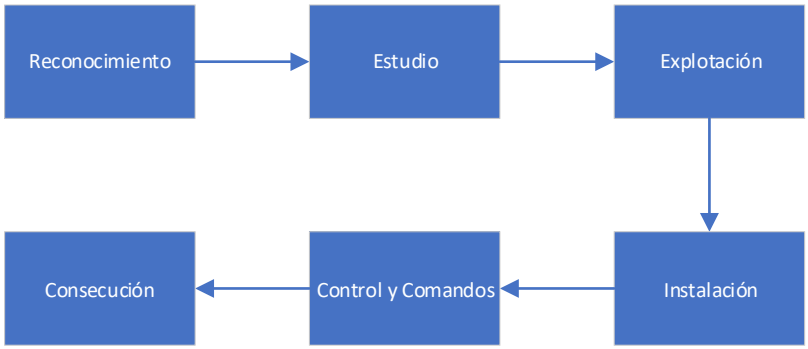

Figura 79 Ciclo de vida de un ataque informático

1. **Reconocimiento**. En la primera fase de un ataque informático, se realizan estudios sobre los objetivos, se detectan y se escogen los más valiosos. En esta fase se suelen dar lugar los ataques de reconocimiento explicados anteriormente.
2. **Estudio**. Los atacantes determinan que métodos deben utilizar para alcanzar su objetivo con mayor eficacia.
3. **Explotación**. En esta etapa, los atacantes explotan una vulnerabilidad en un sistema informático para conseguir entrar en el sistema.
4. **Instalación**. Una vez dentro del sistema, los atacantes instalarán software malicioso con el objetivo de realizar más operaciones, como mantener el acceso posterior y escalar privilegios.
5. **Control y Comandos**. Con el software malicioso ya instalado en el sistema informático objetivo, el atacante puede controlar activamente el sistema estableciendo una comunicación bidireccional con su infraestructura de información para enviar y recibir información del sistema infectado.
6. **Consecución**. Con todo preparado, los atacantes realizan las acciones necesarias para la consecución de su objetivo, como filtración de información, destrucción de infraestructura, extorsión, nuevos ataques informáticos, etc.

2 Honeypot

Dentro del mundo de la detección de intrusiones en red (IDS) donde todos los sistemas de detección se basan en la comparación de eventos a partir de reglas de ataques conocidos

podemos encontrarnos que, ante la evolución de los ataques hacia patrones de comportamiento más complejos, un conjunto de reglas no llega a ser suficiente.

Aquí entran en juego los honeypots. Un honeypot es un sistema informático altamente monitorizado que se configura de manera que sea vulnerable a la mayor cantidad de ataques posible [15]. Un honeypot simula un equipo vulnerable de manera que se puedan detectar tanto nuevos ataques no conocidos (zero days), como nuevas técnicas de explotación de vulnerabilidades. Un honeypot también nos puede servir para distraer la atención de un atacante de sistemas informáticos más valiosos que posea una organización.

Todo el tráfico de red que genera y es dirigido hacia el honeypot será tráfico susceptible de pertenecer o ser consecuencia de un ataque informático en proceso. Los honeypots no resuelven los problemas de seguridad que se puedan producir en otros sistemas informáticos, pero sí ofrecen una gran información sobre ataques informáticos que puede ayudar a la mejora de la protección de otros sistemas informáticos.

2.1 Tipos de honeypots

La clasificación de honeypots pueden ser realizada en dos categorías principales según su nivel de interacción y su finalidad de despliegue [16].

Categorías de clasificación	Tipos de honeypots
Nivel de interacción	Baja interacción
	Media interacción
	Alta interacción
Finalidad de despliegue	Producción
	Investigación

Tabla 19 Clasificación de honeypots

2.1.1 Nivel de interacción

Según el nivel de interacción que proveen los honeypots, se pueden clasificar en: baja interacción, media interacción y alta interacción.

- **Baja interacción**. Los honeypots de baja interacción simulan diferentes servicios de red con la funcionalidad suficiente para que el atacante pueda conectarse e interactuar con ellos, por ejemplo, un servicio FTP.
- **Media interacción**. Los honeypots de media interacción avanzan un paso más que los honeypots de baja interacción y simulan procesos más complejos, como servidores web, servidores de ficheros, etc. También se pueden configurar para almacenar software malicioso.
- **Alta interacción**. Los honeypots de alta interacción son sistemas completos en los que se le da total libertad al atacante para que interactúe con el mientras se realizan todos los procesos de monitorización correspondientes.

2.1.2 Finalidad de despliegue.

Según la finalidad de despliegue de los honeypots se les puede clasificar en: producción e investigación:

- **Honeypots de producción**. Son honeypots que se despliegan dentro de una red en producción con el objetivo de detectar los ataques que se puedan llevar a cabo en

ésta. Estos honeypots se integran dentro de la infraestructura de la organización con el objetivo de detectar amenazas externas e internas a la organización. Este tipo de honeypot suele ser desplegado como honeypot de baja interacción.

- **Honeypots para investigación**. Son honeypots que despliegan expertos en seguridad, normalmente de sombrero blanco, para aprender las herramientas, tácticas y técnicas que utilizan los atacantes. Estos honeypots suelen ser de alta interacción.

3 Honeynets

Una honeynet [17] es un tipo de honeypot basado en una arquitectura de red que contiene un grupo de honeypots interconectados entre sí. El objetivo principal de una honeynet es proporcionar al atacante una red estructurada de sistemas informáticos vulnerables de tal manera que el administrador de la honeynet pueda monitorizar e investigar toda la interacción que el atacante tiene con esta.

Normalmente, los honeypots que pertenecen a una honeynet son honeypots de alta interacción que ofrecen sistemas operativos, servicios y procesos reales para el atacante, con el fin de alcanzar el máximo realismo posible en el proceso de ataque. Todos los honeypots contenidos en una honeynet no tienen ningún fin más allá que el del estudio de técnicas de ataques informáticos, por lo que, cualquier interacción con la honeynet más allá de la del administrador de esta, será susceptible de ser estudiada y considerada como un intento de ataque.

3.1 Requisitos

Para cumplir con las características de una honeynet y mitigar al máximo los riesgos de despliegue de esta, toda arquitectura de red propuesta debe cumplir tres requisitos principales: control de datos, captura de datos y colección de datos.

3.1.1 Control de datos

Principalmente, el control de datos consiste en controlar qué se le permite al atacante hacer y que no se le permite dentro de una honeynet. El mayor riesgo de despliegue de una honeynet es que el atacante utilice la honeynet a su favor para atacar otros sistemas en producción, ya sean de la red interna donde se encuentra la honeynet desplegada o de otra localización en Internet.

El reto principal es limitar las acciones que pueda realizar un atacante sin permitir que dicho atacante detecte que existe un control de datos sobre su interacción con la honeynet. Se le debe permitir al atacante cierto grado de libertad, pero siempre limitando con algún criterio las acciones que se pueden realizar. Mientras más libertad permitamos a un atacante más aprenderemos, pero más riesgo corremos que utilice la honeynet a su favor contra otros sistemas. El balance de libertad y limitación dependerá de cada organización y se adaptará a sus necesidades y posibilidades de asunción de riesgos.

El control de datos se puede implementar mediante la combinación de diferentes mecanismos, como el conteo de conexiones salientes, el ancho de banda utilizado y sistemas de prevención de intrusiones (IPS).

3.1.2 Captura de datos

La captura de datos consiste en monitorizar y recolectar todas las actividades que se llevan a cabo dentro de una honeynet por parte de un atacante. La dificultad radica en captar la máxima cantidad de información sin que el atacante se percate de ello.

La información se debe capturar desde el máximo número de puntos posibles, tanto en la red como en los honeypots. Con esto conseguimos captar información en el máximo número de zonas posible para conseguir una imagen completa de todo el proceso de un ataque informático.

Toda la información que genera un ataque debe ser registrada, ya sean escaneos de puertos, ataques de fuerza bruta, conexiones entrantes y salientes, emisión de comandos en un equipo, descargas de archivos, etc.

3.1.3 Colección de datos

La colección de datos es crucial para proveer al administrador de la honeynet de la posibilidad de analizar y estudiar toda la información recolectada en una honeynet. Toda la información recogida debe ser guardada en un punto centralizado y convenientemente securizado, de tal manera que no es guarde localmente en ningún honeypot y sea susceptible de ser detectada y borrada por el atacante.

Toda la transmisión de información entre los diferentes elementos componentes de una honeynet con el punto de recolección debería producirse mediante canales encriptados, de tal manera que se dificulte su detección por parte del atacante.

3.2 Arquitecturas de despliegue

Según la arquitectura de despliegue de la honeynet y los mecanismos que se implementen en la misma, las honeynets se pueden clasificar en: primera generación, segunda generación y tercera generación.

3.2.1 Primera generación

Una honeynet de primera generación es la primera aproximación a una arquitectura de honeynet que se puede desarrollar. Esta arquitectura consiste en una honeynet desplegada de tal manera que su separación de la red en producción se realiza mediante el despliegue de un cortafuegos al uso con un sistema de detección de intrusiones (IDS).

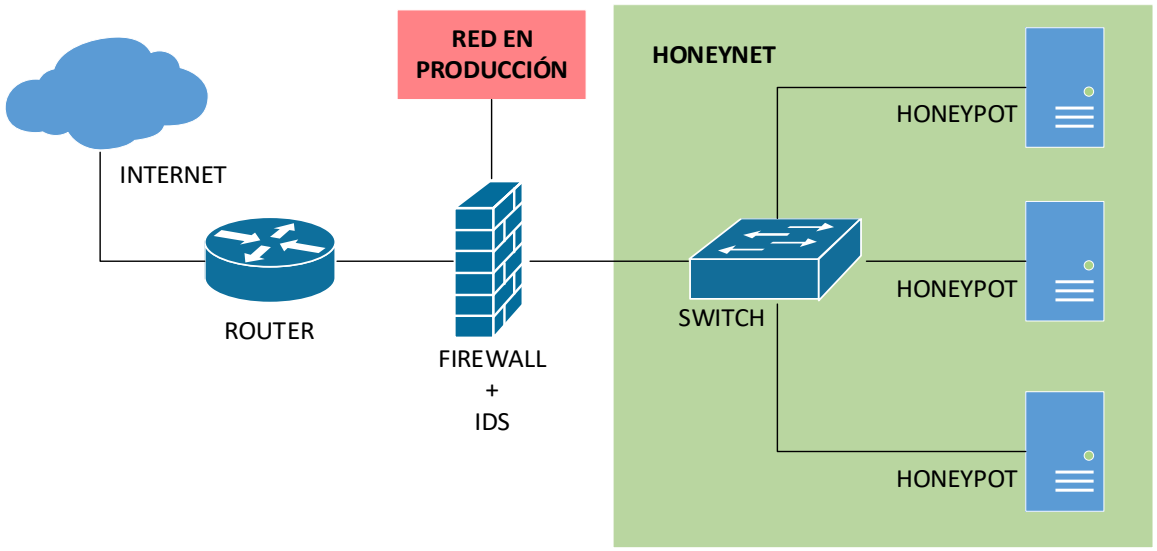

Figura 80 Arquitectura honeynet de primera generación

Esta arquitectura provoca que la honeynet pueda ser fácilmente identificada debido a su separación física y lógica de la red en producción.

3.2.2 Segunda generación

Una honeynet de segunda generación es una honeynet que utiliza como frontera un sistema denominado honeywall. Un honeywall es un dispositivo que actúa como puente entre la honeynet y el resto de la red estructurada de la organización. Dicho honeywall implementa de manera más eficiente el control, la captura y la recolección de los datos que las honeynets de primera generación.

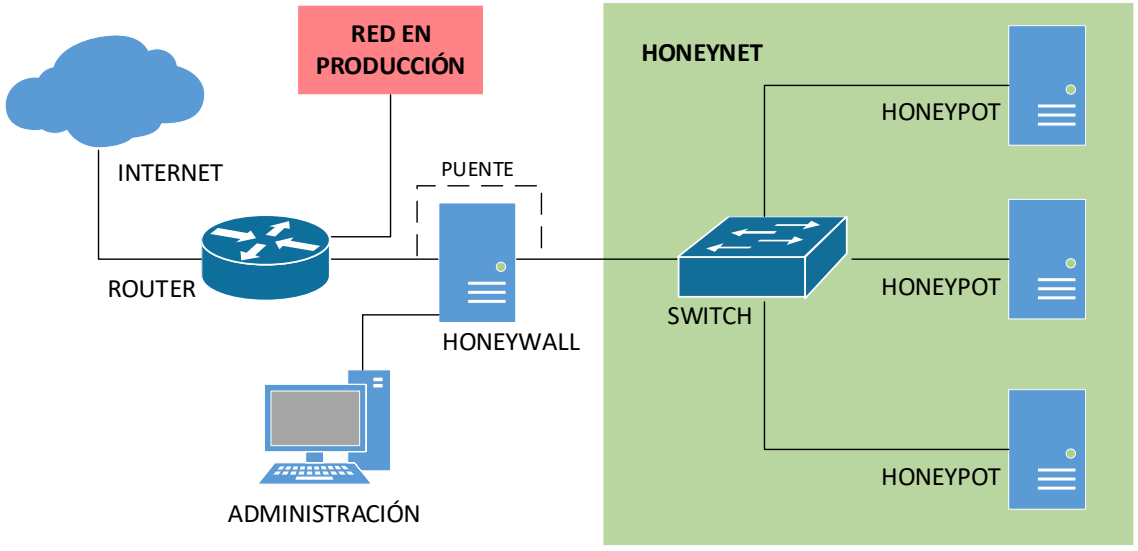

Figura 81 Arquitectura honeynet de segunda generación

Todo el tráfico de red que recibe el honeywall con la honeynet tanto como destino u origen es transportado de manera transparente para los equipos involucrados. Gracias al honeywall disponemos de un punto de monitorización en la red que resulta transparente para el atacante, de manera que es más difícil para él la detección de la honeynet.

Dentro del honeywall se pueden optimizar el control, la captura y la colección de datos de la siguiente manera:

- **Control de datos**. Limitación de conexiones para diferentes protocolos en diferentes escalas de tiempo mediante IPTables. Por ejemplo, se pueden limitar las conexiones TCP a 20 al día, o a 5 el minuto, si se quiere ser más restrictivo.
- **Captura de datos**. Se pueden configurar detectores de intrusión en red (IDS) que monitoricen el puente de red del honeywall y alerten sobre patrones de tráfico sospechosos de ser un ataque informático. También podemos configurar el firewall del honeywall para que se mantenga un registro de conexiones iniciadas, conexiones terminadas, reglas activadas, etc. También podemos configurar en el honeywall algún software que nos genere gráficas de uso de la red para que podamos observar un histórico del tráfico de red de nuestra honeynet.
- **Colección de datos**. Se puede configurar nuestro honeywall para que sea accedido por una interfaz diferente a las dos pertenecientes al puente de red, para que podamos analizar todos los datos capturados por el honeywall en la honeynet y podamos configurar todos los aspectos de la honeynet de los que es responsable el honeywall.

3.2.3 Tercera generación

Las honeynets de tercera generación son honeynets de segunda generación que se integran dentro de una red en producción, de tal manera que, a vista de un atacante, los elementos integrantes de la honeynets estén en la misma red lógica que los equipos en producción, pero siempre controlados y monitorizados gracias al honeywall.

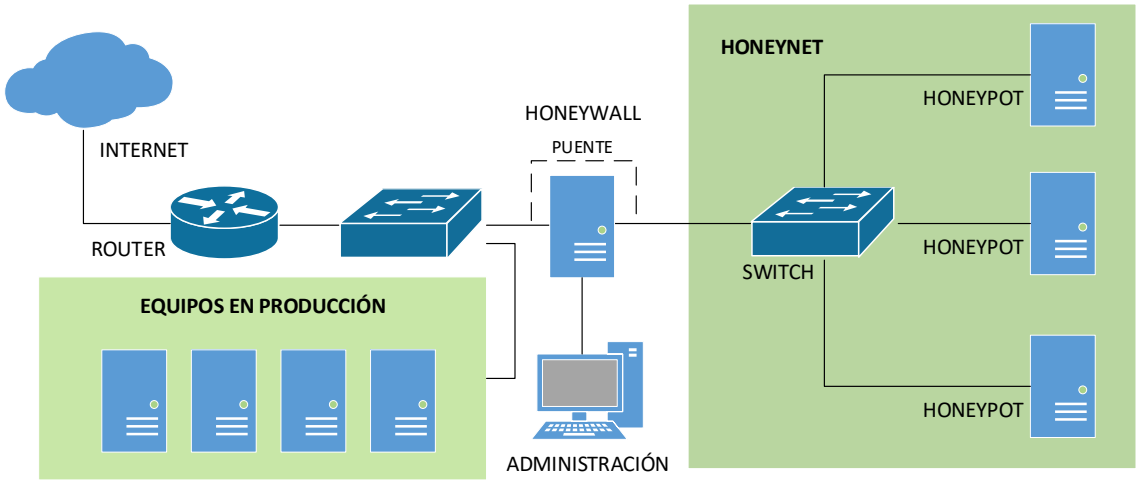

Figura 82 Arquitectura honeynet de tercera generación

De este modo, conseguimos ocultar al máximo la honeynet dentro de la red en producción, para que al atacante le sea lo más difícil posible la tarea de detección de la honeynet. La gran ventaja de una honeynet de tercera generación es que puede resultar de gran ayuda en la detección de ataques informáticos dentro de la red en producción, así como otro tipo de software malicioso en expansión por la red en producción.

3.3 Honeynets virtuales

Gracias a las capacidades de virtualización que ofrecen los sistemas informáticos modernos podemos introducir el concepto de honeynets virtuales. Una honeynet virtual [18] es una honeynet donde los honeypots que la conforman son desplegados mediante software de virtualización. Podemos desplegar una honeynet donde los diferentes honeypots virtuales de alta interacción que ofrecen diferentes servicios vulnerables pueden compartir una misma máquina física, con sus recursos correspondientes (CPU, RAM, etc.).

Dentro del concepto de honeynets virtuales, tenemos dos aproximaciones de despliegue: honeynets virtuales independientes y honeynets virtuales híbridas, cada una de ellas con sus ventajas y desventajas.

Tipos de honeynets virtuales	Ventajas	Desventajas
Independientes	Portables	Punto único de fallo
	Poco coste de despliegue	Altos recursos de computación
Híbridas	Seguridad	Menos portabilidad
	Flexibilidad	Más coste de despliegue

Tabla 20 Tipos de honeynets virtuales

3.3.1 Honeynets virtuales independientes

Una honeynet virtual independiente es aquella en la que todos los honeypots que la conforman, así como el honeywall, si procede, se despliegan mediante software de virtualización en una misma máquina física, compartiendo todos los recursos.

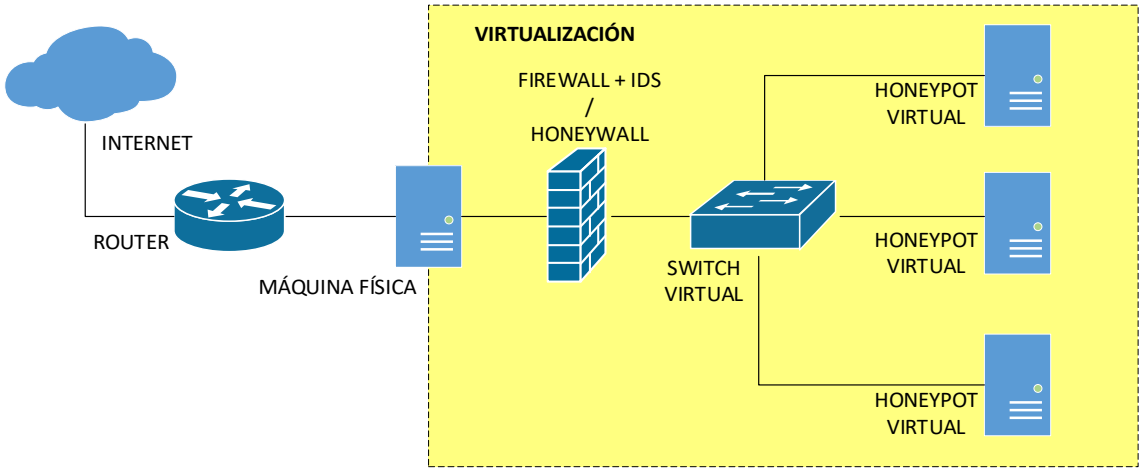

Figura 83 Honeynet virtual independiente

Algunas ventajas son:

- **Portables**. Las honeynets virtuales independientes pueden ser desplegadas en un ordenador portátil y ser transportadas según las necesidades específicas de cada momento para conectarse a la red que sea objeto de estudio.
- **Poco coste de despliegue**. Las honeynets virtuales independientes producen poco coste en cuanto a dinero y espacio, ya que solo se necesita una máquina física para desplegar la honeynet.

Algunas desventajas son:

- **Punto único de fallo**. Al estar desplegada toda la honeynet virtual en una misma máquina física, cualquier fallo de ésta puede provocar que la honeynet virtual deje de funcionar en su totalidad.
- **Altos recursos de computación**. Dependiendo del tamaño de la honeynet virtual, los servicios que presta, y la cantidad de tráfico de red que tiene que soportar, podríamos necesitar una alta capacidad de procesamiento en nuestra máquina física.

3.3.2 Honeynets virtuales híbridas

Una honeynet virtual híbrida es una combinación de las honeynets tradicionales y las honeynets virtuales independientes. Todos los honeypots integrantes de la honeynet son desplegados en una misma máquina física, pero en este caso, el punto de frontera de la honeynet ya sea un firewall + IDS (primera generación) o un honeywall (segunda y tercera generación) se despliegan en una máquina física aislada.

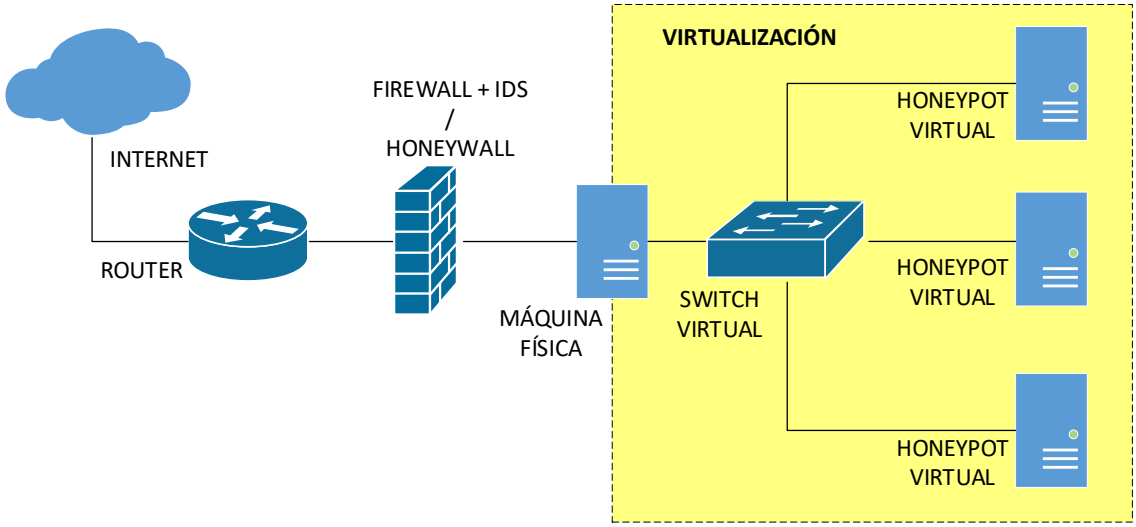

Figura 84 Honeynet virtual híbrida

Algunas ventajas de las honeynets virtuales híbridas son:

- **Seguridad**. Al estar compuesta de más de una misma máquina física, el número de puntos de fallo aumenta, por lo que, si una máquina falla, no tiene por qué perder la honeynet todo su funcionamiento.
- **Flexibilidad.** Al ser el punto de frontera de la red una máquina física separada de la máquina de virtualización de honeypots, se puede adaptar dicho punto más a las necesidades específicas de cada despliegue de la honeynet.

Algunas desventajas de las honeynets virtuales híbridas son:

- **Menos portabilidad**. Al estar compuesta por más de una máquina física, se reduce la facilidad de transportar la honeynet.
- **Más coste de despliegue**. Desplegar una honeynet virtual híbrida implica el uso de más de una máquina física, lo que incurre en el uso de más espacio y en más recursos económicos.

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

REF: 0000002

ANEXO A: CONFIGURACIONES

CLIENTE: EMPRESA PROVINCIAL DE INFORMACIÓN DE CÁDIZ S.A. (EPICSA)
PLAZA MADRID S/N, EDIFICIO CARRANZA,
FONDO SUR, LOCAL 10, 11010 CÁDIZ
956261500

AUTOR: CARLOS CARRETERO AGUILAR
INGENIERO INFORMÁTICO
25603515-F
CARLOS.CARRETEROAGUILAR@ALUM.UCA.ES

FIRMADO:
SOLICITANTE AUTOR

CÁDIZ, A 1 DE JULIO DE 2018

1 Honeywall

1.1 Interfaces de red

Configuración de las interfaces de red del honeywall en /etc/network/interfaces

```
# Interfaz de loopback
auto lo
iface lo inet loopback

# Interfaz puente br0 = eno1 <-> eno2
iface eno1 inet manual
iface eno2 inet manual
auto br0
iface br0 inet static
        address 10.0.0.2
        netmask 255.255.255.0
        broadcast 10.0.0.255
        gateway 10.0.0.1
        bridge_ports eno1 eno2
        bridge_stp off

# Interfaz de administracion
auto eno3
iface eno3 inet static
        address 10.10.10.2
        netmask 255.255.255.0
        broadcast 10.10.10.255
        dns-nameservers 8.8.8.8
        network 10.10.10.0
        gateway 10.10.10.1
```

2 Control de datos

2.1 IPTables

Reglas de IPTables del honeywall, almacenadas en /etc/iptables/rules.v4

```
*nat
:PREROUTING ACCEPT [159:12452]
:INPUT ACCEPT [156:12296]
:OUTPUT ACCEPT [42:2712]
:POSTROUTING ACCEPT [45:2868]
-A PREROUTING -i tun0 -p tcp -m tcp --dport 9999 -j DNAT --to-destination 10.0.0.3:8006
-A POSTROUTING -d 10.0.0.3/32 -p tcp -m tcp --dport 8006 -j SNAT --to-source 10.0.0.2
COMMIT
# Completed on Tue Jun  5 16:01:01 2018
# Generated by iptables-save v1.6.0 on Tue Jun  5 16:01:01 2018
*filter
:INPUT ACCEPT [11196:4958033]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [6849:4652573]
:RATE-LIMIT - [0:0]
-A FORWARD -m conntrack --ctstate NEW,RELATED,ESTABLISHED -j RATE-LIMIT
-A FORWARD -j NFQUEUE --queue-num 0
-A RATE-LIMIT -m hashlimit --hashlimit-upto 50/sec --hashlimit-burst 20 --hashlimit-
mode srcip --hashlimit-name conn_rate_limit -j NFQUEUE --queue-num 0
-A RATE-LIMIT -j LOG --log-prefix "IPTables-Rejected: "
-A RATE-LIMIT -j REJECT --reject-with icmp-port-unreachable
COMMIT
```

2.2 Suricata

Archivo de configuración /etc/suricata/suricata.yml (Se muestra la configuración modificada y/o añadida sobre el archivo por defecto)

```
%YAML 1.1
---

# Suricata configuration file. In addition to the comments describing all
# options in this file, full documentation can be found at:
# https://redmine.openinfosecfoundation.org/projects/suricata/wiki/Suricatayaml

##
## Step 1: inform Suricata about your network
##

vars:
  # more specifc is better for alert accuracy and performance
  address-groups:
    HOME_NET: "[10.0.0.0/24]"
    EXTERNAL_NET: "any"
    HTTP_SERVERS: "10.0.0.4"
    SMTP_SERVERS: "10.0.0.5"

##
## Step 2: select the rules to enable or disable
##

default-rule-path: /etc/suricata/rules
rule-files:
  - local.rules
  #- app-layer-events.rules
  - emerging-deleted.rules
  - emerging-rpc.rules
  - http-events.rules
  - botcc.portgrouped.rules
  - emerging-dns.rules
  - emerging-scada.rules
  - botcc.rules
  - emerging-dos.rules
  - emerging-scan.rules
  #- modbus-events.rules
  - emerging-exploit.rules
  - emerging-shellcode.rules
  - rbn-malvertisers.rules
  - ciarmy.rules
  - emerging-ftp.rules
  - emerging-smtp.rules
  - rbn.rules
  #- emerging-games.rules
  - emerging-snmp.rules
  #- emerging-icmp_info.rules
  - emerging-sql.rules
  - compromised.rules
  #- emerging-icmp.rules
  - emerging-telnet.rules
  - smtp-events.rules
  #- decoder-events.rules
  - emerging-imap.rules
  - emerging-tftp.rules
  #- stream-events.rules
  - dnp3-events.rules
  #- emerging-inappropriate.rules
  - emerging-trojan.rules
  - dns-events.rules
```

```
  - emerging-info.rules
  - emerging-user_agents.rules
  - drop.rules
  - emerging-malware.rules
  - emerging-voip.rules
  - dshield.rules
  - emerging-misc.rules
  - emerging-web_client.rules
  - tls-events.rules
  #- emerging-activex.rules
  - emerging-mobile_malware.rules
  - emerging-web_server.rules
  - tor.rules
  - emerging-attack_response.rules
  - emerging-netbios.rules
  #- emerging-web_specific_apps.rules
  - emerging-chat.rules
  - emerging-p2p.rules
  - emerging-worm.rules
  - emerging-policy.rules
  - emerging-current_events.rules
  - emerging-pop3.rules

classification-file: /etc/suricata/classification.config
reference-config-file: /etc/suricata/reference.config
threshold-file: /etc/suricata/threshold.config


##
## Step 3: select outputs to enable
##

# The default logging directory.  Any log or output file will be
# placed here if its not specified with a full path name. This can be
# overridden with the -l command line parameter.
default-log-dir: /var/log/suricata/

# global stats configuration
stats:
  enabled: no
  # The interval field (in seconds) controls at what interval
  # the loggers are invoked.
  interval: 8

# Configure the type of alert (and other) logging you would like.
outputs:
  # a line based alerts log similar to Snort's fast.log
  - fast:
      enabled: no
      filename: fast.log
      append: yes
      #filetype: regular # 'regular', 'unix_stream' or 'unix_dgram'

  # Extensible Event Format (nicknamed EVE) event log in JSON format
  - eve-log:
      enabled: yes
      filetype: regular #regular|syslog|unix_dgram|unix_stream|redis
      filename: eve.json
      filemode: 644
      #prefix: "@cee: " # prefix to prepend to each log entry
      # the following are valid when type: syslog above
      #identity: "suricata"
      #facility: local5
      #level: Info ## possible levels: Emergency, Alert, Critical,
                   ## Error, Warning, Notice, Info, Debug
      #redis:
```

```
#  server: 127.0.0.1
#  port: 6379
#  mode: list ## possible values: list (default), channel
#  key: suricata ## key or channel to use (default to suricata)
# Redis pipelining set up. This will enable to only do a query every
# 'batch-size' events. This should lower the latency induced by network
# connection at the cost of some memory. There is no flushing implemented
# so this setting as to be reserved to high traffic suricata.
#  pipelining:
#    enabled: yes ## set enable to yes to enable query pipelining
#    batch-size: 10 ## number of entry to keep in buffer
types:
  - alert:
      payload: yes
      payload-buffer-size: 4kb
      payload-printable: yes
      packet: yes
      http: yes               # enable dumping of http fields
      tls: yes                # enable dumping of tls fields
      ssh: yes                # enable dumping of ssh fields
      smtp: yes               # enable dumping of smtp fields

      # HTTP X-Forwarded-For support by adding an extra field or overwriti
      # the source or destination IP address (depending on flow direction)
      # with the one reported in the X-Forwarded-For HTTP header. This is
      # helpful when reviewing alerts for traffic that is being reverse
      # or forward proxied.
      xff:
        enabled: no
        # Two operation modes are available, "extra-data" and "overwrite".
        mode: extra-data
        # Two proxy deployments are supported, "reverse" and "forward". In
        # a "reverse" deployment the IP address used is the last one, in a
        # "forward" deployment the first IP address is used.
        deployment: reverse
        # than one IP address is present, the last IP address will be the
        # one taken into consideration.
        header: X-Forwarded-For
  - http:
      extended: yes     # enable this for extended logging information
      # custom allows additional http fields to be included in eve-log
      # the example below adds three additional fields when uncommented
      #custom: [Accept-Encoding, Accept-Language, Authorization]
  - dns
  #- tls:
  #    extended: yes     # enable this for extended logging information
  - files:
      force-magic: no   # force logging magic on all logged files
      force-md5: no     # force logging of md5 checksums
  #- drop:
  #    alerts: no       # log alerts that caused drops
  - smtp:
      extended: yes # enable this for extended logging information
      custom:[received, x-mailer, x-originating-ip, relays, reply-to,bcc
 ,subject]
      md5: [body, subject]

  - ssh
  - stats:
      totals: yes       # stats for all threads merged together
      threads: no       # per thread stats
      deltas: no        # include delta values
  # bi-directional flows
  #- flow
  # uni-directional flows
  #- netflow
```

```
    #copy-mode: ips
    #copy-iface: eth1
```

Configuración de Oinkmaster para la actualización automática de reglas en /etc/oinkmaster.conf

```
# $Id: oinkmaster.conf,v 1.134 2008/02/18 19:33:45 andreas_o Exp $ #

url = http://rules.emergingthreats.net/open/suricata/emerging.rules.tar.gz
modifysid * "^alert (.*classtype\s*:\s*attempted-admin)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*attempted-user)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*inappropriate-content)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*shellcode-detect)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*successful-admin)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*successful-user)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*trojan-activity)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*unsuccessful-user)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*web-application-attack)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*attempted-dos)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*attempted-recon)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*unusual-client-port-connection)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*system-call-detect)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*suspicious-login)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*suspicious-filename-detect)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*successful-recon-limited)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*successful-recon-largescale)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*successful-dos)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*rpc-portmap-decode)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*non-standard-protocol)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*misc-attack)" | "drop ${1}"
modifysid * "^alert (.*classtype\s*:\s*denial-of-service)" | "drop ${1}"
```

Configuración de tarea CRON para la actualización diaria de reglas en /etc/cron.daily/suricata-rule-update

```
# ls -l /etc/cron.daily/suricata-rule-update
-rwxr-xr-x 1 root root 83 may 22 12:03 /etc/cron.daily/suricata-rule-update


# cat /etc/cron.daily/suricata-rule-update
oinkmaster -C /etc/oinkmaster.conf -o /etc/suricata/rules
service suricata restart
```

Configuración de rotación de logs de suricata en /etc/logrotate.d/suricata

```
/var/log/suricata/*.log /var/log/suricata/*.json
{
    rotate 3
    missingok
    nocompress
    create
    sharedscripts
    postrotate
            /bin/kill -HUP `cat /var/run/suricata.pid 2>/dev/null` 2>/dev/null || true
    endscript
}
```

Configuración del servicio demonio de Suricata para su continua ejecución en /etc/systemd/system/suricata.service

```
[Unit]
```

```
Description=Suricata IDS/IDP daemon
After=network.target
Requires=network.target
Documentation=man:suricata(8) man:suricatasc(8)
Documentation=https://redmine.openinfosecfoundation.org/projects/suricata/wiki

[Service]
Type=forking
Environment=LD_PREDLOAD=/usr/lib/libtcmalloc_minimal.so.4
Environment=CFG=/etc/suricata/suricata.yaml PID=/var/run/suricata.pid
CapabilityBoundingSet=CAP_NET_ADMIN
PIDFile=/var/run/suricata.pid
ExecStart=/usr/bin/suricata -D -q 0 -c $CFG --pidfile $PID -D
ExecReload=/bin/kill -HUP $MAINPID
ExecStop=/bin/kill $MAINPID
PrivateTmp=yes
InaccessibleDirectories=/home /root
ReadOnlyDirectories=/boot /usr /etc

[Install]
WantedBy=multi-user.target
```

3 Captura de datos

3.1 OSSEC

Configuración añadida en el fichero de configuración /var/ossec/etc/ossec.conf

```
# File: /var/ossec/etc/ossec.conf
<ossec_config>
  <global>
    <jsonout_output>yes</jsonout_output>
    ...
  </global>
  ...
</ossec_config>

<directories check_all="yes" realtime="yes">/etc,/usr/bin,/usr/sbin</directories>
<directories check_all="yes" realtime="yes" >/bin,/sbin,/boot</directories>
<directories check_all="yes" realtime="yes">/home</directories>

<alerts>
   <log_alert_level>1</log_alert_level>
</alerts>
```

Reglas de decodificación añadidas a OSSEC para decodificar eventos para el honeypot FTP y el honeypot SSH en /var/ossec/etc/decoder.xml

```
<decoder name="ssh-invalid-user">
  <parent>sshd</parent>
  <prematch>^Invalid user|^Illegal user</prematch>
  <regex offset="after_prematch"> from (\S+)$</regex>
  <order>srcip</order>
</decoder>

<decoder name="ssh-invfailed">
  <parent>sshd</parent>
  <prematch>^Failed \S+ for invalid user|^Failed \S+ for illegal user</prematch>
  <regex offset="after_prematch">from (\S+) port \d+ \w+$</regex>
  <order>srcip</order>
</decoder>

<decoder name="vsftpd_invuser">
```

```
  <parent>vsftpd</parent>
  <regex offset="after_parent">[(\S+)] FTP response: Client "(\S+\w)", "530 Permission
denied."$</regex>
  <order>user,srcip</order>
</decoder>
```

Reglas añadidas a OSSEC para la detección de eventos en el honeypot FTP en /var/ossec/rules/vsftp.log

```
 <rule id="11403" level="5">
    <if_sid>11400</if_sid>
    <match>FAIL LOGIN: </match>
    <description>Login failed accessing the FTP server.</description>
    <group>authentication_failed,</group>
  </rule>

  <rule id="11404" level="5">
    <if_sid>11400</if_sid>
    <match>OK UPLOAD: </match>
    <description>FTP server file upload.</description>
  </rule>

  <rule id="11405" level="5">
    <if_sid>11400</if_sid>
    <match>OK DOWNLOAD: </match>
    <description>FTP server file download.</description>
  </rule>

  <rule id="11406" level="5">
    <if_sid>11400</if_sid>
    <match>OK DELETE: </match>
    <description>FTP server file deletion.</description>
  </rule>
<rule id="11407" level="5">
       <decoded_as>vsftpd_invuser</decoded_as>
    <if_sid>11400</if_sid>
    <match>530 Permission denied.</match>
    <description>FTP failed user</description>
</rule>
```

3.2 Tcpdump

Comando tcpdump escrito en /etc/rc.local para su ejecución como demonio al inicio del honeywall

```
tcpdump -i br0 -w /var/log/tcpdump/trace-%Y-%m-%d_%H.%M.pcap -W 48 -G 3600 -C 2048 -K
-n &
```

3.3 Fprobe

Configuración de fprobe en /etc/default/fprobe después de su instalación

```
#fprobe default configuration file

INTERFACE="br0"
FLOW_COLLECTOR="localhost:2055"

#fprobe can't distinguish IP packet from other (e.g. ARP)
OTHER_ARGS="-fip"
```

3.4 Collectd

Configuración añadida al fichero de configuración /etc/collectd/collectd.conf

```
LoadPlugin cpu
LoadPlugin memory
LoadPlugin disk
LoadPlugin network

<Plugin cpu>
       ValuesPercentage true
</Plugin>

<Plugin memory>
        ValuesAbsolute false
        ValuesPercentage true
</Plugin>

<Plugin "disk">
       Disk "sda"
       Disk "/^hd/"
       IgnoreSelected false
</Plugin>

<Plugin network>
       <Server "127.0.0.1" "25826"></Server
</Plugin>
```

3.5 Proxmox

Configuración en /etc/pve/status.cfg

```
graphite:
   server 10.0.0.2 # IP HW
   port 2003
   path proxmox
```

3.6 Copia de logs de honeypots

Para la copia de ficheros de logs del honeypot HP1 en el honeywall, se utiliza la herramienta rsync en el honeywall con el siguiente script, previa compartición de clave pública del honeywall en el honeypot HP1 en /root/.ssh/authorized_keys

```
root@honeywall# cat /root/sync_logs_hp1.sh

#!/bin/bash
rsync -avz -e "ssh -o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null"
root@10.0.0.4:/var/log/ /var/log/hp1
chmod -R a+r var/log/hp1
```

Mediante una tarea cron, se ejecuta el script cada minuto. Para ello, se añade la siguiente linea en el fichero /etc/crontab

```
*  *    * * *   root    /root/sync-logs-hp1.sh
```

4 Colección de datos

4.1 Elasticsearch

La configuración de elasticsearch en /etc/elasticsearch/elasticsearch.yml es la siguiente

```
network.host: localhost
http.port: 9200
```

4.2 Logstash

Las diferentes configuraciones de logstash para la colección de los datos de las diversas fuentes en /etc/logstash/conf.d/ es la siguiente:

- Suricata: suricata.conf

```
input {
  file {
    path => ["/var/log/suricata/eve.json"]
    sincedb_path => ["/var/lib/logstash/since.db"]
    codec =>   json
    type => "SuricataIDPS"
  }
}

filter {
  if [type] == "SuricataIDPS" {
    date {
      match => [ "timestamp", "ISO8601" ]
    }
    ruby {
      code => "if event.get('event_type') == 'fileinfo';
event.get('fileinfo')['type']=event.get('fileinfo')['magic'].to_s.split(',')[0];
end;"
    }
  }

  if [src_ip] {
    mutate {
      add_field => { "src_ip_rdns" => "%{src_ip}" }
    }
    dns {
      reverse => [ "src_ip_rdns" ]
      action => "replace"
    }
  }
  if [dest_ip] {
    mutate {
      add_field => { "dest_ip_rdns" => "%{dest_ip}" }
    }
    dns {
      reverse => [ "dest_ip_rdns" ]
      action => "replace"
    }
  }
}

output {
  if [type] == "SuricataIDPS" {
  elasticsearch {
    hosts => localhost
    index => "logstash-suricata-%{+YYYY.MM.dd}"
  }
 }
}
```

- OSSEC: ossec.conf

```
input {
   file {
       type => "ossec"
       path => "/var/ossec/logs/alerts/alerts.json"
       codec => "json"
   }
```

```
}
output {
    if [type] == "ossec" {
       elasticsearch {
          hosts => localhost
           index => "logstash-ossec-%{+YYYY.MM.dd}"
       }
       #stdout { codec => rubydebug }
   }
}
```

- Fprobe: fprobe.conf

```
input {
  udp {
    port  => 2055
    codec => netflow
    type  => netflow
  }
}
filter {
  if [type] == "netflow" {
    if [netflow][protocol] == 1 {
      mutate {
        add_field => { "protocol" => "ICMP" }
      }
    }
    else if [netflow][protocol] == 2 {
      mutate {
        add_field => { "protocol" => "IGMP" }
      }
    }
    else if [netflow][protocol] == 6 {
      mutate {
        add_field => { "protocol" => "TCP" }
      }
    }
    else if [netflow][protocol] == 17 {
      mutate {
        add_field => { "protocol" => "UDP" }
      }
    }
    else if [netflow][protocol] == 27 {
      mutate {
        add_field => { "protocol" => "RDP" }
      }
    }
  }
}
output {
  if [type] == "netflow" {
    elasticsearch {
      hosts => localhost
      index => "logstash-netflow-%{+YYYY.MM.dd}"
    }
  }
}
```

- Collectd: collectd.conf

```
input {
  udp {
    port => 25826
    buffer_size => 1452
    codec => collectd {}
```

```
    type => collectd
  }
}

output {
  if [type] == "collectd" {
    elasticsearch {
      hosts => localhost
      index => "logstash-collectd-%{+YYYY.MM.dd}"
    }
  }
}
```

- Proxmox: proxmox.conf

```
input {
  udp {
    port => 2003
    buffer_size => 1452
    codec => graphite {}
    type => proxmox
  }
}

output {
  if [type] == "proxmox" {
    elasticsearch {
      hosts => localhost
      index => "logstash-proxmox-%{+YYYY.MM.dd}"
    }
  }
}
```

- Servidor Apache en el honeypot HP1

```
input {
  file {
    path => "/var/log/hp1/apache2/access.log"
    #start_position => "beginning"
    type => "hp1-apache"
  }
}

filter {
if [type] == "hp1-apache" {
  if [path] =~ "access" {
    grok {
      match => { "message" => "%{COMBINEDAPACHELOG}" }
    }
  }
  date {
    match => [ "timestamp" , "dd/MMM/yyyy:HH:mm:ss Z" ]
  }
}
}

output {
  if [type] == "hp1-apache" {
    if "_grokparsefailure" not in [tags] {
       elasticsearch {
          hosts => localhost
          index => "logstash-hp1-apache-%{+YYYY.MM.dd}"
       }
   }
  }
```

```
}
```

5 Análisis de datos y administración

5.1 OpenVPN

Fichero de configuración del servidor en /etc/openvpn/server.conf

```
port 1194
proto udp
dev tun
sndbuf 0
rcvbuf 0
ca ca.crt
cert server.crt
key server.key
dh dh.pem
auth SHA512
tls-auth ta.key 0
topology subnet
server 10.8.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
push "redirect-gateway def1 bypass-dhcp"
push "dhcp-option DNS 8.8.8.8"
keepalive 10 120
cipher AES-256-CBC
comp-lzo
user nobody
group nogroup
persist-key
persist-tun
status openvpn-status.log
verb 3
crl-verify crl.pem
```

Fichero de configuración de cliente VPN

```
client
dev tun
dev-type tun
dev-node tun0
proto udp
sndbuf 0
rcvbuf 0
remote 83.37.127.151 1194
resolv-retry infinite
nobind
remote-cert-tls server
auth SHA512
cipher AES-256-CBC
comp-lzo
key-direction 1
verb 3s
route-nopull
<ca>
-----BEGIN CERTIFICATE-----
```

... Claves omitidas

5.2 SSH

Configuración añadida al servidor SSH en /etc/ssh/sshd.config

```
ListenAddress 10.8.0.1
```

5.3 Kibana

Configuración añadida al fichero de configuración de Kibana en /etc/kibana/kibana.yml

```
server.ssl.enabled: true
server.ssl.key: /etc/kibana/certs/kibana-selfsigned.key
server.ssl.certificate: /etc/kibana/certs/kibana-selfsigned.crt
server.host: 10.8.0.1
kibana.defaultAppId: "dashboard/a1663770-63f8-11e8-807b-11cfef214f0b"
```

6 Software de honeypots

6.1 Honeypot HP1

6.1.1 Características de la máquina virtual

En el servidor Proxmox, se crea una máquina virtual con las siguientes características:

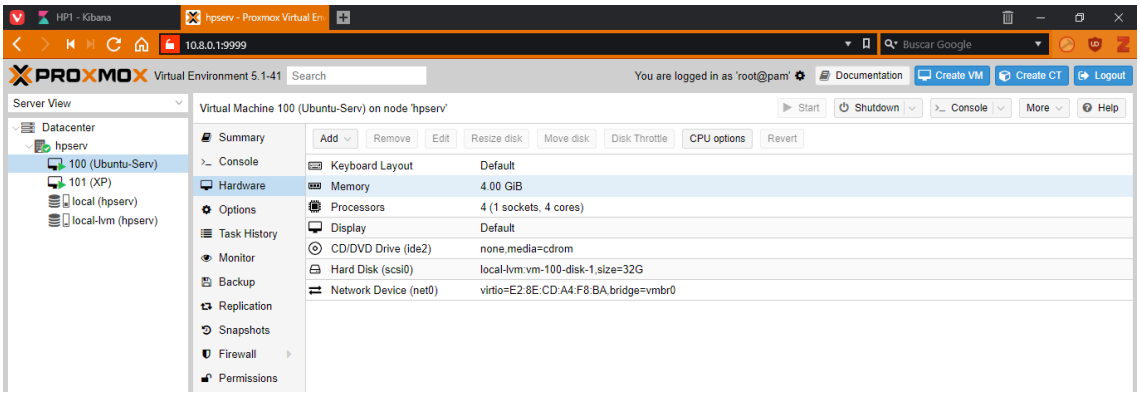

Figura 85 Características de máquina virtual de honeypot HP1

6.1.2 Preparación del sistema

Se añaden las siguientes líneas al fichero /etc/sudoers para que los usuarios con bajos privilegios consigan privilegios de administración con las herramientas sudo y su

```
root    ALL=(ALL:ALL) ALL
#%admin ALL=(ALL) ALL
#%sudo  ALL=(ALL:ALL) ALL
```

Se eliminan los permisos de ejecución para los usuarios no administradores de los binarios del sistema que den información sobre usuarios conectados en el sistema

```
root@hp1# chmod o-x /usr/bin{who, w, users}
```

Se eliminan los permisos de lectura y modificación para los usuarios no administradores del directorio /var del honeypot, para evitar el borrado de logs por parte de un atacante

```
root@hp1# chmod -R o-rwx /var
```

6.1.3 Servidor Web

La instalación se realiza por defecto con

```
# apt-get install apache2
```

6.1.4 Servidor FTP

La configuración aplicada al honeypot HP1 previo a la instalación del servidor FTP es

```
# apt-get install vsftpd
# cp /etc/vsftpd.conf /etc/vsftpd.conf.orig
# adduser test
# chown nobody:nogroup /home/test
# chmod a-w /home/test
# mkdir /home/test/files
# chown test:test /home/test/files
# echo "vsftpd test file" | sudo tee /home/test/files/test.txt
# echo "test" >> /etc/vsftpd.userlist
# echo "test" >> /etc/vsftpd.chroot_list
```

La configuración añadida al fichero de configuración del software vsftpd es

```
listen=YES
listen_ipv6=NO
local_enable=YES
write_enable=YES
syslog_enable=YES
xferlog_enable=YES
xferlog_std_format=NO
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd.chroot_list
user_sub_token=$USER
local_root=/home/$USER
userlist_enable=YES
userlist_file=/etc/vsftpd.userlist
userlist_deny=NO
log_ftp_protocol=YES
```

6.1.5 Servidor SSH

Servidor SSH instalado por defecto con

```
# apt install openssh-server
```

6.1.6 Registro de comandos

En el fichero /etc/rc.local se añade la siguiente línea, para el registro de comandos en cada inicio de sesión

```
test "$(ps -ocommand= -p $PPID | awk '{print $1}')" == 'script' || (script -q -f
/var/log/script/script.$(date -u +%Y-%m-%d-%H-%M-%S).${HOSTNAME:-
$(hostname)}.$USER.log)
```

6.1.7 Cliente OSSEC

Configuración añadida al fichero de configuración de OSSEC en /var/ossec/etc/ossec.config

```
<client>
  <server-ip>10.0.0.2</server-ip>
</client>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/auth.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
```

```
  <location>/var/log/vsftpd.log</location>
</localfile>
```

6.2 Honeypot HP2

6.2.1 Características de la máquina virtual

En el servidor Proxmox, se crea una máquina virtual con las siguientes características:

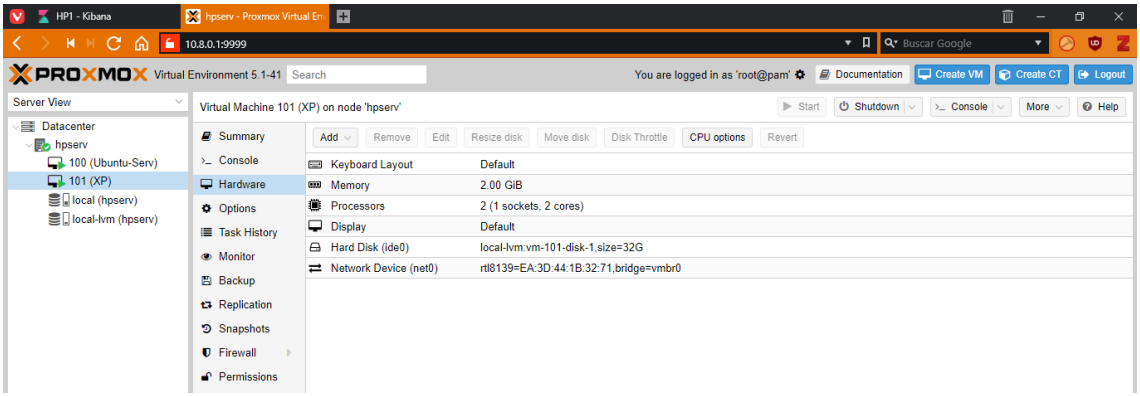

Figura 86 Características de máquina virtual de honeypot HP2

6.2.2 Servidor SMTP

La configuración del servidor SMTP en Windows XP SP3 es la que se realiza por defecto en la instalación del servidor IIS en Windows. La instalación se realiza en Panel de control > Agregar o quitar programas > Agregar o quitar componentes de Windows, seleccionando Servicio de Internet Information Service (IIS) y aplicando los cambios.

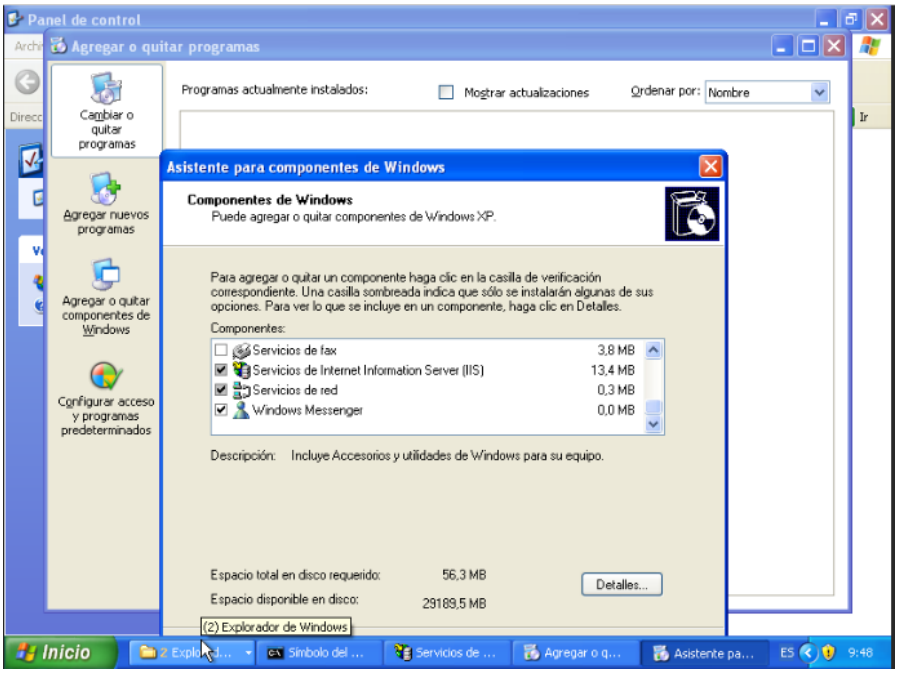

Figura 87 Instalando Servidor IIS en Windows XP SP3

Por defecto, el servidor SMTP ya viene instalado, tal y como se puede observar en la siguiente imagen.

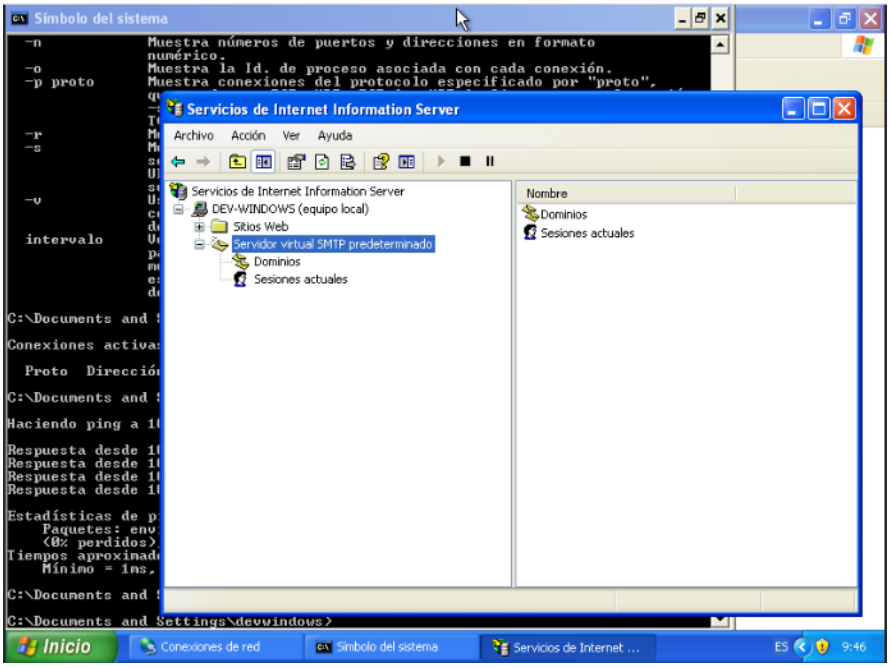

Figura 88 Servidor SMTP en Windows XP SP3

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

REF: 0000002

ESPECIFICACIONES DEL SISTEMA

CLIENTE: EMPRESA PROVINCIAL DE INFORMACIÓN DE CÁDIZ S.A. (EPICSA)
PLAZA MADRID S/N, EDIFICIO CARRANZA,
FONDO SUR, LOCAL 10, 11010 CÁDIZ
956261500

AUTOR: CARLOS CARRETERO AGUILAR
INGENIERO INFORMÁTICO
25603515-F
CARLOS.CARRETEROAGUILAR@ALUM.UCA.ES

FIRMADO:
SOLICITANTE AUTOR

CÁDIZ, A 1 DE JULIO DE 2018

1 Objetivos del proyecto

Los objetivos del proyecto de despliegue de una honeynet virtual para la investigación de ataques informáticos en la red de la Diputación de Cádiz son los siguientes:

OBJ-01	Arquitectura de despliegue
Descripción	La honeynet debe tener una arquitectura de despliegue de segunda o tercera generación

Tabla 21 Objetivo del proyecto 01

OBJ-02	Virtualización
Descripción	La honeynet debe desplegarse mediante software virtualizado

Tabla 22 Objetivo del proyecto 02

OBJ-03	Control de datos
Descripción	La honeynet debe implementar todos los mecanismos necesarios de control de datos

Tabla 23 Objetivo del proyecto 03

OBJ-04	Captura de datos
Descripción	La honeynet debe implementar todos los mecanismos necesarios de captura de datos

Tabla 24 Objetivo del proyecto 04

OBJ-05	Colección de datos
Descripción	La honeynet debe implementar todos los mecanismos necesarios de colección de datos

Tabla 25 Objetivo del proyecto 05

OBJ-06	Análisis de datos
Descripción	La honeynet debe implementar todas las herramientas necesarias para el análisis óptimo de los datos

Tabla 26 Objetivo del proyecto 06

2 Requisitos del proyecto

Los objetivos del proyecto de despliegue de una honeynet virtual para la detección y el estudio de ataques informáticos son los siguientes:

- **R-01:** Desplegar una honeynet que no sobrecargue la infraestructura existente.
- **R-02:** Desplegar una honeynet escalable.
- **R-03:** Implementar control y limitación de conexiones.
- **R-04:** Implementar prevención de intrusiones en red.
- **R-05:** Implementar detección de intrusiones en honeypots.
- **R-06**: Implementar control de integridad en honeypots.
- **R-07:** Implementar registro de inicios de sesión.
- **R-08:** Implementar registro de comandos.
- **R-09:** Implementar captura de tráfico.
- **R-10:** Recoger estadísticas de uso y sesiones de la red**.**
- **R-11:** Recoger estadísticas de uso de recursos de la honeynet.
- **R-12:** Implementar cifrado en la colección de datos.
- **R-13:** Implementar un sistema de análisis de información.
- **R-14:** Configurar una conexión cifrada con el honeywall.

R-01	**Desplegar una honeynet que no sobrecargue la infraestructura existente**
Descripción	La arquitectura de la honeynet desplegada no debe sobrecargar la infraestructura de red en producción
Datos asociados	- No procede

Tabla 27 Requisito del proyecto 01

R-02	**Desplegar una honeynet escalable**
Descripción	La arquitectura de la honeynet debe permitir la rápida y fácil escalabilidad de los servicios desplegados
Datos asociados	- No procede

Tabla 28 Requisito del proyecto 02

R-03	**Implementar control y limitación de conexiones**
Descripción	La honeynet debe controlar y limitar todas las sesiones establecidas con honeypots de la red vulnerable
Datos asociados	- IP y puerto de origen de la sesión - IP y puerto de destino de la sesión - Protocolo de red - Número de paquetes de red intercambiados - Número de bytes transmitidos en la sesión

Tabla 29 Requisito del proyecto 03

R-04	**Implementar prevención de intrusiones en red**
Descripción	La honeynet debe ser capaz de alertar y bloquear ante la detección de tráfico de red sospechoso
Datos asociados	- IP y puerto de origen del tráfico - IP y puerto de destino del tráfico - Fecha de creación del registro - Tipo del posible ataque

Tabla 30 Requisito del proyecto 04

R-05	**Implementar detección de intrusiones en honeypots**
Descripción	La honeynet debe ser capaz de alertar ante la detección de software malicioso en los honeypots
Datos asociados	- Honeypot involucrado - Descripción de la actividad sospechosa - Fecha de detección de la actividad

Tabla 31 Requisito del proyecto 05

R-06	**Implementar control de integridad en honeypots**
Descripción	La honeynet debe ser capaz de alertar ante el cambio de ficheros o software existentes en el sistema, así como la aparición de nuevos ficheros o software
Datos asociados	- Honeypot involucrado - Descripción del fichero/software modificado o aparecido - Fecha de detección de la actividad

Tabla 32 Requisito del proyecto 06

R-07	**Implementar registro de inicios de sesión**
Descripción	La honeynet debe capturar todos los registros de inicio de sesión en los diferentes honeypots

Datos asociados	- Honeypot involucrado - Descripción del inicio de sesión - Fecha de detección de la actividad

Tabla 33 Requisito del proyecto 07

R-08	**Implementar registro de comandos**
Descripción	La honeynet debe capturar todos los registros de comandos emitidos en los diferentes honeypots
Datos asociados	- Honeypot involucrado - Comando emitido - Fecha de emisión del comando

Tabla 34 Requisito del proyecto 08

R-09	**Implementar captura de tráfico**
Descripción	La honeynet debe capturar todo el tráfico que la atraviese
Datos asociados	- No procede

Tabla 35 Requisito del proyecto 09

R-10	**Recoger estadísticas de uso y sesiones de la red**
Descripción	La honeynet debe generar estadísticas de uso de la red con información de las diferentes sesiones establecidas
Datos asociados	- Equipos involucrados - Ancho de banda consumido - Fecha de inicio de la sesión

Tabla 36 Requisito del proyecto 10

R-11	**Recoger estadísticas de uso de recursos de la honeynet**
Descripción	La honeynet debe generar estadísticas de uso de los recursos de los honeypots y el honeywall
Datos asociados	- Honeypots involucrados - Recursos consumidos

Tabla 37 Requisito del proyecto 11

R-12	**Implementar cifrado en la colección de datos**
Descripción	Todo intercambio de información entre el honeywall y los honeypots debe ser a través de canales cifrados
Datos asociados	- No procede

Tabla 38 Requisito del proyecto 12

R-13	**Implementar un sistema de análisis de información**
Descripción	La honeynet debe provisionar de un sistema de análisis de información residente en el honeywall
Datos asociados	- No procede

Tabla 39 Requisito del proyecto 13

R-14	**Configurar una conexión cifrada con el honeywall**
Descripción	Toda conexión que se establezca con el honeywall ya sea para análisis de datos o administración de la honeynet, debe ser cifrada
Datos asociados	- No procede

Tabla 40 Requisito del proyecto 14

3 Matriz de rastreabilidad

Mediante el cruce de los objetivos para el despliegue de una honeynet virtual para la detección y análisis de ataques informáticos con los requisitos para su óptimo y correcto funcionamiento, se procede a la generación de la matriz de rastreabilidad de objetivos/requisitos.

Requisitos / Objetivos	01	02	03	04	05	06	07	08	09	10	11	12	13	14
01	•													
02		•												
03			•	•										
04					•	•	•	•	•	•				
05											•	•		
06													•	•

Tabla 41 Matriz de rastreabilidad de objetivos/requisitos

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

REF: 0000002

MEDICIONES

CLIENTE: EMPRESA PROVINCIAL DE INFORMACIÓN DE CÁDIZ S.A. (EPICSA)
PLAZA MADRID S/N, EDIFICIO CARRANZA,
FONDO SUR, LOCAL 10, 11010 CÁDIZ
956261500

AUTOR: CARLOS CARRETERO AGUILAR
INGENIERO INFORMÁTICO
25603515-F
CARLOS.CARRETEROAGUILAR@ALUM.UCA.ES

FIRMADO:
SOLICITANTE AUTOR

CÁDIZ, A 1 DE JULIO DE 2018

1 Cableado

Cable	Unidades
Latiguillo RJ-45 FTP Cat. 5e LSZH (1 metro)	2
Latiguillo RJ-45 FTP Cat. 5e LSZH (5 metros)	2

Tabla 42 Mediciones de cableado

2 Conexionado a Internet

Conexión	Meses
ADSL 12 Mb/s para administración	3

Tabla 43 Mediciones de conexionado a Internet

3 Hardware

Hardware	Unidades
HPE ProLiant DL360 Gen9 Intel Xeon E5-2620v4	2

Tabla 44 Mediciones de hardware

4 Personal

Personal	Horas
Autor del proyecto	175

Tabla 45 Mediciones de personal

DESPLIEGUE DE UNA HONEYNET EN LA RED DE LA DIPUTACIÓN DE CÁDIZ PARA LA INVESTIGACIÓN DE ATAQUES INFORMÁTICOS

REF: 0000002

PRESUPUESTO

CLIENTE: EMPRESA PROVINCIAL DE INFORMACIÓN DE CÁDIZ S.A. (EPICSA)
PLAZA MADRID S/N, EDIFICIO CARRANZA,
FONDO SUR, LOCAL 10, 11010 CÁDIZ
956261500

AUTOR: CARLOS CARRETERO AGUILAR
INGENIERO INFORMÁTICO
25603515-F
CARLOS.CARRETEROAGUILAR@ALUM.UCA.ES

FIRMADO:
SOLICITANTE AUTOR

CÁDIZ, A 1 DE JULIO DE 2018

1 Cableado

Cable	Unidades	Precio/unidad (€)	Precio total (€)
Latiguillo RJ-45 FTP Cat. 5e LSZH (1 metro)	2	1,80	3,60
Latiguillo RJ-45 FTP Cat. 5e LSZH (5 metros)	2	2,52	5,04
		Total:	**8,64**

Tabla 46 Presupuesto de cableado

2 Conexionado a Internet

Conexión	Meses	Precio/mes (€)	Precio total (€)
ADSL 12 Mb/s para administración	3	35.90	107.70
		Total:	**107.70**

Tabla 47 Presupuesto de conexionado a Internet

3 Hardware

Hardware	Unidades	Precio/unidad (€)	Precio total (€)
HPE ProLiant DL360 Gen9 Intel Xeon E5-2620v4	2	2.197,54	4.395,08
		Total:	**4.395,08**

Tabla 48 Presupuesto de hardware

4 Personal

Personal	Horas	Precio/hora (€)	Precio total (€)
Autor del proyecto	175	12,03	2.105,25
		Total:	**2.105,25**

Tabla 49 Presupuesto de personal

5 Total

	Precio (€)
Cableado	8,64
Conexionado a Internet	107.70
Hardware	4.395,08
Autor del proyecto	2.105,25
Total:	**6.616,67**

Tabla 50 Presupuesto total